Editora da Universidade Federal de Campina Grande

Autômatos Finitos

com uma introdução aos
Autômatos Finitos Quânticos

Elloá B. Guedes
Bernardo Lula Jr.

Este livro foi desenvolvido na Universidade Federal de Campina Grande como um projeto do Instituto de Estudos em Computação e Informação Quânticas, IQuanta, e tem por objetivo oferecer a alunos e pesquisadores em Ciência da Computação e áreas afins uma introdução aos Autômatos Finitos, desde os modelos mais simples até as mais recentes publicações sobre os modelos quânticos e suas variantes.

A apresentação dos autômatos é feita de forma clara, didática, com ilustrações, exemplos e provas de teoremas. O livro apresenta também um conjunto de exercícios ao final de cada capítulo com o objetivo de promover uma melhor fixação dos conceitos apresentados. Há ainda um conjunto de referências bibliográficas atualizadas para aqueles que desejarem aprofundar-se no tema.

EDUFCG
EDITORA DA UNIVERSIDADE FEDERAL DE CAMPINA GRANDE

978-85-89674-83-6
9 788589 674836

Autômatos Finitos

com uma introdução aos Autômatos Finitos Quânticos

Elloá B. Guedes
Departamento de Sistemas e Computação
Instituto de Estudos em Computação e Informação Quânticas
Universidade Federal de Campina Grande

Bernardo Lula Júnior
Departamento de Sistemas e Computação
Instituto de Estudos em Computação e Informação Quânticas
Universidade Federal de Campina Grande

EDUFCG - Editora da Universidade Federal de Campina Grande
Campina Grande - Paraíba
2009

Editora da Universidade Federal de Campina Grande

EXPEDIENTE

Prof. Thompson Fernandes Mariz

Reitor

Prof. Dr. Edílson Amorim

Vice-Reitor

Prof. Dr. Antonio Clarindo Barbosa de Souza

Diretor Administrativo da EDUFCG

Prof. Dr. Antonio Gomes da Silva

Diretor Comercial da EDUFCG

Conselho Editorial da EDUFCG:

Prof. Benedito Antonio Luciano – CEEI
Prof. Carlos Alberto Vieira de Azevedo – CTRN
Profª Consuelo Padilha Vilar – CCBS
Prof. Joaquim Cavalcante Alencar – CCJS (Sousa)
Prof. José Helder Pinheiro – CH
Prof. Onaldo Guedes Rodrigues – CSTR (Patos)
Prof. Warderley Alves de Sousa – CFP (Cajazeiras)

Autores
Elloá Barreto Guedes da Costa
Bernardo Lula Júnior

Ilustração
Elloá Barreto Guedes da Costa

Diagramação
Marconi Luiz França

Capa
Elloá Barreto Guedes da Costa
Marconi Luiz França

Campina Grande – 2009

http://www.ufcg.edu.br/~edufcg
edufcg@reitoria.ufcg.edu.br

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL DA UFCG

C837a Costa, Elloá B. Guedes da.
Autômatos finitos: com uma introdução autômatos finitos quânticos / Elloá B. Guedes da Costa, Bernardo Lula Júnior. • Campina Grande, EDUFCG, 2009.
145p.

ISBN 978-85-89674-83-6

1. Ciência da Computação. 2. Física. 3. Autômatos. 4. Física Quântica. I. Título.

CDU 531.18

"*Existe uma máscara de teoria encobrindo toda a natureza.*"

William Whewell

(1794, 1866)

Sumário

Prefácio

A Teoria da Computação caracteriza-se como uma importante área de estudo da Ciência da Computação, pois nela estudam-se os fundamentos que descrevem o computador como um modelo matemático. Esta teoria foi proposta em 1936 por Alan Turing, que definiu um modelo matemático de computação conhecido como *máquina de Turing*. Segundo Turing, este modelo seria capaz de capturar o significado de se realizar uma tarefa por meio de algoritmos. Isto é, se um algoritmo pode ser realizado em qualquer tipo de sistema físico, então existe um algoritmo equivalente para uma máquina de Turing universal que realiza exatamente a mesma tarefa que o algoritmo original. Esta afirmação, conhecida como a tese de Church-Turing, teve sua evidência fortalecida a cada novo modelo que ia sendo proposto e que tinha sua equivalência com o modelo de Turing demonstrada. Seguindo a larga aceitação desta tese pela comunidade científica, foi observado que o modelo de computação da máquina de Turing era ao menos tão poderoso quanto qualquer outro modelo de computação, no sentido de que um problema que podia ser resolvido eficientemente em algum modelo de computação também poderia ser resolvido eficientemente no modelo

da máquina de Turing.

Em meados de 1970, Robert Solovay e Volker Strassen verificaram que a combinação de uma máquina de Turing com elementos probabilísticos, tais como um gerador de números aleatórios, seria capaz de resolver eficientemente problemas que não têm solução eficiente no modelo original de Turing, essencialmente determinístico. Este novo modelo ficou conhecido como *máquina de Turing probabilística* e ampliou o conceito de que qualquer processo algorítmico poderia ser simulado eficientemente por uma máquina de Turing probabilística, a chamada Tese Forte de Church-Turing.

Em 1982 o físico Richard Feynman conjecturou que nenhuma máquina de Turing era capaz de simular fenômenos quânticos de forma eficiente. Para tanto, era necessário um modelo computacional capaz de levar em consideração os princípios da Mecânica Quântica. Este modelo foi então proposto em 1985 por David Deutsch, sendo denominado *máquina de Turing quântica.* O modelo proposto por Deutsch desencadeou desafios à tese forte de Church-Turing, particularmente quando algoritmos quânticos eficientes para os problemas da fatoração (algoritmo de Shor) e de busca não estruturada (algoritmo de Grover) foram propostos.

Paralelamente ao desenvolvimento da Teoria da Computação, nos moldes descrito acima, outra teoria conexa estava sendo desenvolvida e que veio a se chamar *Teoria dos Autômatos.* Os principais conceitos da Teoria dos Autômatos foram introdu-

zidos nas décadas de 40 e 50 como resultados do esforço de diversos pesquisadores incluindo matemáticos, lingüistas, neurofisiologistas e engenheiros eletricistas. Os trabalhos iniciais da área buscavam o desenvolvimento de máquinas que modelassem os processos cognitivos do cérebro humano, tendo, neste sentido, como primeiro expoente o próprio modelo de Turing, que baseou seu trabalho na modelagem do processo de prova de teorema utilizado pelos matemáticos.

Em 1943, McCulloch, psiquiatra, e Pitts, matemático, construíram um modelo para explicar o funcionamento dos neurônios utilizando o conceito de máquinas de estado finito. Em 1951, Kleene publicou seu trabalho no qual introduziu o conceito de linguagens regulares. Huffman, em 1954, apresentou a noção de estado de um autômato e de tabela de transição. Moore, em 1956, apresentou um algoritmo de minimização. Os conceitos de autômatos finitos não-determinísticos e probabilísticos foram introduzidos por Rabin e Scott em 1959 que expuseram os principais conceitos da área de forma sistemática.

No contexto da Teoria da Computação, o modelo de *Autômato Finito* é um dos modelos de computação mais simples, porém com vasta aplicação nas áreas de processadores de texto, compiladores, projetos de circuitos digitais, processamento de linguagem natural, etc. Por causa da simplicidade e do interesse prático, o estudo deste modelo de computação em sala de aula, como componente teórico dos cursos de Computação, é de fundamental importância para o ensino da matéria pois possibilita

a prática de definições formais, proporciona o aprendizado de conceitos relevantes e favorece o aprendizado de modelos mais complexos, como a própria máquina de Turing.

De acordo com Ambainis e Freivals, é possível que as primeiras implementações de computadores quânticos não sejam compostas inteiramente de componentes baseados na Mecânica Quântica. Ao invés disso, acredita-se que existirá a parte quântica, a parte clássica e uma forma de comunicação entre elas. Em um primeiro momento, em virtude da inovação e da baixa escalabilidade, a parte quântica será mais cara e mais reduzida que a parte clássica. Isto implica que serão necessários modelos computacionais que façam uso da Mecânica Quântica, mas que ao mesmo tempo sejam simples e passíveis de implementação física, o que sugere e motiva o estudo de autômatos finitos quânticos, modelo computacional que faz uso da Mecânica Quântica e é o análogo quântico dos autômatos finitos citados acima, os quais fazem uso da Física Clássica.

Este livro, concebido como um projeto de iniciação científica na Universidade Federal de Campina Grande (UFCG) e desenvolvido no Instituto de Estudos em Computação e Informação Quânticas (IQuanta) objetiva oferecer aos alunos dos cursos de graduação em Ciência da Computação um texto que introduza de forma clara e didática os conceitos de Computação Quântica a partir dos conhecimentos que os alunos já possuem, utilizando a analogia com os conceitos clássicos conhecidos como ferramenta pedagógica. Segundo Moore e Crutchfield, quando se

busca entender computação num contexto quântico, pode ser útil trasladar tantos conceitos da teoria da computação clássica quantos forem possíveis para o caso quântico, a começar do nível mais baixo da hierarquia computacional (hierarquia de Chomsky), ou seja, dos autômatos finitos. Um problema adicional neste processo de entendimento é, porém, a necessidade de conhecimentos teóricos em Física, Matemática e Computação que não fazem parte do conteúdo normalmente explorado na graduação em Ciência da Computação.

Assim, neste livro, é apresentada, em dois capítulos (Capítulo 1 e Capítulo 2), uma revisão dos conceitos de Matemática (Álgebra Linear e Vetorial Complexa) e de Física (Mecânica Quântica) básicos necessários para o entendimento do assunto. Em seguida, no Capítulo 3, é feita uma revisão dos modelos de autômatos finitos determinísticos e não-determinísticos. O Capítulo 4 é dedicado a uma introdução ao modelo de autômato probabilístico, que geralmente não faz parte do conteúdo ministrado sobre autômatos nos cursos de graduação em Ciência da Computação, mas sua compreensão é fundamental para o entendimento de autômatos finitos quânticos. Por fim, no Capítulo 5 é apresentado os modelos básicos de autômatos finitos quânticos propostos na literatura sobre o assunto. Embora estes modelos possuam um forte caráter matemático, procurou-se direcionar esforços em busca do aprendizado dos conceitos de maneira simples e intuitiva, com exercícios evidenciando as definições e afirmações. Provas que exijam um conhecimento maior

por parte dos alunos são deixadas para uma seção de exercícios propostos, para aqueles que realmente desejarem prosseguir na busca de conhecimento sobre Computação Quântica.

Os autores agradecem a Cheyenne Ribeiro G. Isidro, por ter compartilhado diversas referências bibliográficas bastante úteis na escrita deste trabalho, Gustavo Jansen, pela revisão dos capítulos iniciais desta obra, e a Carla Souza e Marçal Targino, pela colaboração na produção da capa deste livro. Agradecimentos especiais à UFCG e ao IQuanta pelo apoio ao projeto, que viabilizaram a realização deste trabalho. Os autores esperam que esta publicação estimule o interesse no assunto entre os estudantes dos cursos de graduação em Ciência da Computação e ajude a popularizar a Computação Quântica no País.

Campina Grande, Agosto de 2009
Elloá B. Guedes e Bernardo Lula Júnior

Lista de símbolos

$\mathbb{C}$	Conjunto dos números complexos
i	Unidade Imaginária
$\mathbb{R}$	Conjunto dos números reais
$\|\cdot\rangle$	Notação de Dirac para representar um vetor
$\langle\cdot\|$	Notação de Dirac para representar o dual de um vetor
$\langle\cdot\|\cdot\rangle$	Notação de Dirac para representar o produto interno
$\|\cdot\rangle\langle\cdot\|$	Notação de Dirac para representar o produto externo
$\otimes$	Produto Tensorial
$\oplus$	Soma módulo 2
$\dagger$	Notação para o conjugado transposto
Q	Conjunto de estados de um autômato
Σ	Alfabeto de entrada de um autômato
δ	Função de transição de um autômato
F	Conjunto de estados finais de um autômato
\$	Marcador de término de uma palavra
λ	Palavra vazia
Q_{ac}	Conjunto de estados de aceitação
Q_{rej}	Conjunto de estados de rejeição
Q_{non}	Conjunto de estados de não-parada
Ω	Alfabeto da fita auxiliar

Capítulo 1

Noções Gerais de Álgebra Linear

Este capítulo apresenta as noções gerais de Álgebra Linear necessárias ao aprendizado dos conceitos da Computação Quântica. Este conteúdo costuma ser abordado superficialmente em disciplinas de Álgebra Linear dos cursos de Ciência da Computação; portanto, para cada conceito novo a ser apresentado será feita uma breve revisão a respeito do mesmo. Os exemplos ao longo do capítulo e os exercícios extras podem ser utilizados como ferramentas úteis para fixar o conhecimento apresentado.

1.1 Espaço de Hilbert

A Mecânica Quântica caracteriza-se como um *framework*[1] matemático voltado para a Física Moderna que leva em consideração os efeitos quânticos [18, 33].

A Mecânica Quântica é descrita em termos de um espaço vetorial, denominado *Espaço de Hilbert*, que possui duas carac-

[1]Entende-se por *framework* uma estrutura de suporte. Neste caso, em particular, um conjunto de regras matemáticas que dão suporte às leis da Física.

terísticas: é um espaço vetorial complexo de dimensão finita e possui produto interno.

1.1.1 Primeira Característica: O espaço de Hilbert é um espaço vetorial complexo de dimensão finita

Um espaço vetorial de dimensão n é um conjunto de todas as n-tuplas $(z_1, z_2, \ldots, z_n)$, denominadas "vetores" e cada z_j é um valor numérico (inteiro, real, complexo, etc.).

A representação de um vetor também se dá segundo uma matriz-coluna:

$$(z_1, z_2, \ldots, z_n) \equiv \begin{bmatrix} z_1 \\ z_2 \\ \ldots \\ z_n \end{bmatrix}$$

No caso de um espaço vetorial complexo, cada z_j é um número complexo ($z_j \in \mathbb{C}$). Isto significa que podem ser escritos na forma:

$$z_j = a + b \cdot i$$

em que $a, b \in \mathbb{R}$, i é a unidade imaginária e $i^2 = -1$.

Denota-se por z_j^* o conjugado complexo de z_j. Se z_j é um número complexo da forma $z_j = a \pm b \cdot i$. O seu conjugado complexo z_j^* é da forma $z_j = a \mp b \cdot i$.

Geralmente a Mecânica Quântica é descrita utilizando-se a *notação de Dirac* para representar vetores [12]. Esta notação é

uma forma concisa de representação dos conceitos da Mecânica Quântica, que acarreta em um simplificação dos cálculos a serem realizados.

Nesta notação, um vetor é denominado *ket* e denotado por $|\psi\rangle$, em que ψ é um rótulo e $|\cdot\rangle$ indica que o objeto em questão é um vetor.

Utilizando a notação de Dirac, um vetor em um espaço vetorial $\mathbb{C}^n$ pode ser representado por:

$$|\psi\rangle \equiv \begin{bmatrix} z_1 \\ z_2 \\ \cdots \\ z_n \end{bmatrix}$$

Define-se por *bra* o conjugado transposto de um vetor. O bra de um vetor $|\psi\rangle$ é denotado por $\langle\psi|$ em que ψ é um rótulo e $\langle\cdot|$ indica que o objeto em questão é um bra.

Um bra é uma matriz-linha, obtida da seguinte forma:

$$\begin{aligned} \langle\psi| &= (|\psi\rangle^*)^T \\ &= \begin{bmatrix} z_1^* \\ z_2^* \\ \cdots \\ z_n^* \end{bmatrix}^T \\ &= \begin{bmatrix} z_1^* & z_2^* & \cdots & z_n^* \end{bmatrix} \end{aligned}$$

Exemplo 1.1: Denote a matriz-linha correspondente ao $\langle\varphi|$ do vetor

$$|\varphi\rangle = \begin{bmatrix} 3+2\cdot i \\ 4-i \end{bmatrix}$$

A matriz-linha desejada é obtida da seguinte forma:

$$\langle\varphi| = (|\varphi\rangle^*)^T = \left(\begin{bmatrix} 3+2\cdot i \\ 4-i \end{bmatrix}^* \right)^T$$

$$= \begin{bmatrix} 3-2\cdot i \\ 4+i \end{bmatrix}^T = \begin{bmatrix} 3-2\cdot i & 4+i \end{bmatrix}.$$

Deste modo, a matriz-linha desejada é $\begin{bmatrix} 3-2\cdot i & 4+i \end{bmatrix}$.

Qualquer vetor de um espaço vetorial pode ser descrito como uma combinação linear de vetores de uma base deste espaço. Uma base B é formada pelos seguintes vetores:

$$B = \{|\psi_0\rangle, |\psi_1\rangle, \ldots, |\psi_{n-1}\rangle\} = \sum_{i=0}^{n-1} |\psi_i\rangle$$

Para todo vetor $|\psi\rangle$ de um espaço vetorial de dimensão n e um escalar $\lambda \in \mathbb{C}$, $|\psi\rangle$ pode ser descrito como a seguinte combinação linear:

$$|\psi\rangle = \sum_{i=0}^{n-1} \lambda_i \cdot |\psi_i\rangle$$

Exemplo 1.2: Seja $B = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$ uma base de um espaço vetorial $\mathbb{C}^3$. Denote o vetor

$$|\psi\rangle = \begin{bmatrix} 4 + 5 \cdot i \\ 5 - i \\ i \end{bmatrix}$$

como uma combinação linear dos vetores da base B.

Para denotar o vetor $|\psi\rangle$ como uma combinação linear dos vetores da base B, é necessário satisfazer à seguinte igualdade, em que $|b_i\rangle$ é um vetor pertencente à base B:

$$|\psi\rangle = \sum_{i=0}^{3} \lambda_i \cdot |b_i\rangle$$

Para satisfazer à igualdade, é necessário encontrar os valores de λ_i. Para obtê-los, é necessário prosseguir o desenvolvimento da equação:

$$
\begin{aligned}
|\psi\rangle &= \lambda_1 |b_1\rangle + \lambda_2 |b_2\rangle + \lambda_3 |b_3\rangle \\
&= \lambda_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \lambda_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\end{aligned}
$$

$$
\begin{bmatrix} 4+5\cdot i \\ 5-i \\ i \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix}
$$

Deste modo, os valores de λ_1, λ_2 e λ_3 que satisfazem à igualdade são, respectivamente, $4+5\cdot i$, $5-i$ e i.

Portanto, o vetor $|\psi\rangle$ pode ser escrito como uma combinação linear dos vetores da base B, da seguinte forma:

$$
|\psi\rangle = (4+5\cdot i) \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + (5-i) \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + i \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
$$

1.1.2 Segunda Característica: O espaço de Hilbert é dotado de um produto interno

A segunda característica de um espaço de Hilbert é a existência de um *produto interno*.

O produto interno é uma métrica que relaciona o tamanho de um vetor em relação a outro vetor do mesmo espaço [26]. É

definido como uma função de dois vetores de um mesmo espaço vetorial que tem como valor um número complexo.

Utilizando a notação de Dirac, no espaço de Hilbert o produto interno de dois vetores $|v\rangle$ e $|w\rangle$ possui a seguinte definição:

$$\begin{aligned}\langle v|\, w\rangle &= \begin{bmatrix} v_1^* & \dots & v_n^* \end{bmatrix} \begin{bmatrix} w_1 \\ \dots \\ w_n \end{bmatrix} \\ &= [v_1^* \cdot w_1 + \ldots + v_n^* \cdot w_n]\end{aligned}$$

Exemplo 1.3: Qual o valor do produto interno $\langle\psi|\, \varphi\rangle$? Em que $|\psi\rangle$ e $|\varphi\rangle$ são dados a seguir:

$$|\psi\rangle = \begin{bmatrix} 0 \\ i \end{bmatrix} \quad |\varphi\rangle = \begin{bmatrix} 2+i \\ -i \end{bmatrix}$$

O valor do produto interno desejado pode ser obtido da seguinte forma:

$$\begin{aligned}\langle\psi|\, \varphi\rangle &= \begin{bmatrix} 0 & -i \end{bmatrix} \begin{bmatrix} 2+i \\ -i \end{bmatrix} \\ &= \begin{bmatrix} 0 \cdot (2+i) + (-i) \cdot (-i) \end{bmatrix} = [i^2] = [-1] \\ &= -1.\end{aligned}$$

O valor do produto interno $\langle \psi | \varphi \rangle$ é igual a -1.

A definição do produto interno apresentada satisfaz três requisitos:

1. É linear no segundo argumento:

$$\begin{aligned}
\langle v | w \rangle &= \begin{bmatrix} v_1^* & \dots & v_n^* \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \cdot w_1 \\ \dots \\ \lambda_n \cdot w_n \end{bmatrix} \\
&= [v_1^* \cdot \lambda_1 \cdot w_1 + \ldots + v_n^* \cdot \lambda_n \cdot w_n] \\
&= \sum_{i=1}^{n} \lambda_i \cdot [v_1^* \cdot w_1 + \ldots + v_n^* \cdot w_n] \\
&= \sum_{i=1}^{n} \lambda_i \langle v | w_i \rangle
\end{aligned}$$

2. Resulta no conjugado complexo quando a ordem dos vetores for invertida (comutatividade conjugada):

$$\begin{aligned}
\langle v | w \rangle &= [w_1^* \cdot v_1 + \ldots + w_n^* \cdot v_n] \\
&= ([w_1 \cdot v_1^* + \ldots + w_n \cdot v_n^*])^* \\
&= \langle w | v \rangle^*
\end{aligned}$$

3. É maior ou igual zero quando os vetores forem idênticos (positividade):

$$\begin{aligned}
\langle v | v \rangle &= [v_1^* \cdot v_1 + \ldots + v_n^* \cdot v_n] \\
&\geq 0
\end{aligned}$$

O produto interno de um vetor por ele mesmo só é nulo quando o vetor é o vetor nulo.

O primeiro conceito decorrente da definição de produto interno é o de *norma* de um vetor. A norma de um vetor $|v\rangle$ é denotada por $|||v\rangle||$ e é dada por:

$$|||v\rangle|| \equiv \sqrt{\langle v|\, v\rangle}$$

Quando a norma de um vetor é igual a 1, diz-se que este vetor é *unitário*.

Exemplo 1.4: Qual a norma do vetor $|\psi\rangle$?

$$|\psi\rangle = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$$

A norma do vetor $|\psi\rangle$ pode ser obtida como se segue:

$$\begin{aligned} |||\psi\rangle|| &= \sqrt{\langle\psi|\, \psi\rangle} \\ &= \sqrt{\begin{bmatrix} 3 & 4 \end{bmatrix} \begin{bmatrix} 3 \\ 4 \end{bmatrix}} \\ &= \sqrt{25} = 5. \end{aligned}$$

Conclui-se que a norma do vetor $|\psi\rangle$ é igual a 5.

Para vetores que não são unitários, o processo de normalização permite que a partir de um destes vetores seja definido um vetor de norma unitária. A normalização é feita dividindo-se o vetor por sua norma. Por exemplo, a normalização do vetor $|\psi\rangle$ é denotada abaixo:

$$\frac{|\psi\rangle}{|||\psi\rangle||}$$

Exemplo 1.5: O vetor $|\varphi\rangle = \begin{bmatrix} 1 \\ i \end{bmatrix}$ é um vetor unitário? Em caso contrário, obtenha um novo vetor a partir da normalização do $|\varphi\rangle$.

Para afirmar se o vetor $|\varphi\rangle$ é unitário, é necessário que se conheça a norma deste vetor. Assim:

$$\begin{aligned} |||\varphi\rangle|| &= \sqrt{\langle\varphi|\,\varphi\rangle} \\ &= \sqrt{1-i^2} = \sqrt{1+1} = \sqrt{2}. \end{aligned}$$

A norma do vetor $|\varphi\rangle$ é igual a $\sqrt{2}$, portanto, este vetor não é unitário. Seja o vetor $|\varphi'\rangle$ o vetor obtido a partir da normalização do vetor $|\varphi\rangle$. $|\varphi'\rangle$ é da forma:

$$
\begin{aligned}
|\varphi'\rangle &= \frac{|\varphi\rangle}{\sqrt{2}} \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{-i}{\sqrt{2}} \end{bmatrix}.
\end{aligned}
$$

A norma do vetor $|\varphi'\rangle$ é igual a 1. Isto pode ser comprovado através da obtenção do valor de sua norma:

$$
\begin{aligned}
\big|\big||\varphi'\rangle\big|\big| &= \sqrt{\langle\varphi'|\,\varphi'\rangle} \\
&= \sqrt{\frac{1}{2} + \frac{-i^2}{2}} \\
&= \sqrt{\frac{1+1}{2}} = \sqrt{\frac{2}{2}} \\
&= \sqrt{1} = 1.
\end{aligned}
$$

Isso significa que o vetor $|\varphi'\rangle$ é um vetor unitário.

Quando o produto interno entre dois vetores é igual a zero, diz-se que estes vetores são *ortogonais* entre si.

A caracterização do produto interno consolida a segunda característica de um espaço de Hilbert. O estudo destes espaços vetoriais é importante, pois estes são os espaços de interesse para a Computação e Informação Quânticas.

Exemplo 1.6: Os vetores $|\psi\rangle$ e $|\varphi\rangle$ são ortogonais entre si? Estes dois vetores são dados a seguir:

$$|\psi\rangle = \begin{bmatrix} i \\ 1 \end{bmatrix} \quad |\varphi\rangle = \begin{bmatrix} 1 \\ i \end{bmatrix}$$

Para responder ao questionamento, é necessário verificar qual o valor do produto interno entre estes dois vetores. Caso este seja nulo, os vetores são ortogonais.

O produto interno $\langle\psi|\,\varphi\rangle$ pode ser obtido da seguinte forma:

$$\begin{aligned} \langle\psi|\,\varphi\rangle &= \begin{bmatrix} -i & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ i \end{bmatrix} \\ &= [-i + i] = 0. \end{aligned}$$

Como o valor do produto interno $\langle\psi|\,\varphi\rangle$ é nulo, é possível concluir que os vetores $|\psi\rangle$ e $|\varphi\rangle$ são ortogonais entre si.

1.2 Produto Tensorial

O *produto tensorial* de dois espaços vetoriais V_m e W_n, denotado por $V_m \otimes W_n$, produz um novo espaço vetorial cuja dimensão é $m \times n$. Esta operação é responsável pela definição de um espaço de Hilbert expandido, capaz de ser utilizado para representar sistemas multi-partículas.

Exemplo 1.7: Sejam dois espaços vetoriais de números com-

plexos V_5 e W_7. Qual a dimensão do espaço $V_5 \otimes W_7$?
O valor da dimensão do espaço $V_5 \otimes W_7$ é obtido a partir do produto das dimensões dos espaços V e W, ou seja, $5 \, x \, 7 = 35$. Assim, a dimensão do espaço vetorial obtido é igual a 35.

Os vetores pertencentes ao espaço $V_m \otimes W_n$ são combinações lineares de produtos tensoriais $|v\rangle \otimes |w\rangle$, em que $|v\rangle \in V$ e $|w\rangle \in W$.

Por exemplo, seja o espaço de Hilbert bidimensional. O produto tensorial deste espaço com ele próprio, ou seja, $\mathbb{C}^2 \otimes \mathbb{C}^2$, resulta em um novo espaço vetorial cujos vetores são da forma:

$$\begin{bmatrix} \alpha \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \\ \beta \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \alpha \cdot \alpha \\ \alpha \cdot \beta \\ \beta \cdot \alpha \\ \beta \cdot \beta \end{bmatrix}$$

Exemplo 1.8: Sejam os vetores $|\psi\rangle = \begin{bmatrix} i \\ -i \end{bmatrix}$ e $|\varphi\rangle = \begin{bmatrix} 3 \\ 2 \cdot i \end{bmatrix}$. Qual a representação matricial do vetor $|\psi\rangle \otimes |\varphi\rangle$? Dado que $|\psi\rangle$ e $|\varphi\rangle$ são dados a seguir:

$$|\psi\rangle = \begin{bmatrix} i \\ -i \end{bmatrix} \quad |\varphi\rangle = \begin{bmatrix} 3 \\ 2 \cdot i \end{bmatrix}$$

A representação matricial do vetor $|\psi\rangle \otimes |\varphi\rangle$ pode ser obtida como se segue:

$$|\psi\rangle \otimes |\varphi\rangle = \begin{bmatrix} i\begin{bmatrix} 3 \\ 2 \end{bmatrix} \\ -i\begin{bmatrix} 3 \\ 2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 3 \cdot i \\ 2 \cdot i \\ -3 \cdot i \\ -2 \cdot i \end{bmatrix}.$$

O vetor $|\psi\rangle \otimes |\varphi\rangle$ pertence a um espaço vetorial de dimensão 4, obtido a partir do produto tensorial do espaço V_2 com ele mesmo, em que $|\psi\rangle, |\varphi\rangle \in V$.

Utilizando a Notação de Dirac, o produto tensorial dos vetores $|\psi\rangle$ e $|\varphi\rangle$ pode ser denotado por $|\psi\rangle \otimes |\varphi\rangle$ ou ainda por $|\psi\rangle |\varphi\rangle$. No caso particular do tensorial de um vetor por ele mesmo, basta utilizar a notação $|\psi\rangle^{\otimes k}$, em que k denota o número de vezes que o produto tensorial é efetuado [12].

Exemplo 1.9: Seja V um espaço vetorial complexo de dimensão igual a 3. Qual a dimensão do espaço $V^{\otimes 6}$?

A notação $V^{\otimes 6}$ indica que os seguintes produtos tensoriais são efetuados:

$$V^{\otimes 6} = V \otimes V \otimes V \otimes V \otimes V \otimes V$$

O valor numérico da dimensão do espaço vetorial $V^{\otimes 6}$ pode ser obtido da seguinte forma, em que $dim()$ é uma função que tem como valor a dimensão do espaço vetorial:

$$\begin{aligned} dim(V^{\otimes 6}) &= dim(V) \times \ldots \times dim(V) \\ &= 3 \times 3 \times 3 \times 3 \times 3 \times 3 \\ &= 3^6 = 729. \end{aligned}$$

Assim, a dimensão do espaço vetorial $V^{\otimes 6}$ é igual a 729. É interessante notar o aumento da dimensão do espaço vetorial diante do pequeno número de produtos tensoriais realizados.

O produto tensorial deve satisfazer as propriedades abaixo, considerando λ um escalar arbitrário, $|v\rangle$,$|v_1\rangle$ e $|v_2\rangle$ vetores do espaço vetorial V e $|w\rangle$,$|w_1\rangle$ e $|w_2\rangle$ vetores do espaço W:

1. Ser linear para escalares:

$$\begin{aligned} |v\rangle \otimes (\lambda \cdot |w\rangle) &= \lambda \cdot (|v\rangle\,|w\rangle) \\ (\lambda \cdot |v\rangle) \otimes |w\rangle &= \lambda \cdot (|v\rangle\,|w\rangle) \end{aligned}$$

2. Ser linear para vetores, em duas situações:

   (a) $|v\rangle \otimes (|w_1\rangle + |w_2\rangle) = |v\rangle\,|w_1\rangle + |v\rangle\,|w_2\rangle$

   (b) $(|v_1\rangle + |v_2\rangle) \otimes |w\rangle = |v_1\rangle\,|w\rangle + |v_2\rangle\,|w\rangle$

Exemplo 1.10: Utilizando as três propriedades que o produto tensorial deve satisfazer, re-escreva a expressão $(|v\rangle \otimes \lambda |w\rangle) \otimes ((|a\rangle + |b\rangle) \otimes |c\rangle)$, em que $|v\rangle, |w\rangle, |a\rangle, |b\rangle, |c\rangle$ são vetores quaisquer e λ é uma constante.
A primeira propridade pode ser aplicada ao trecho $|v\rangle \otimes \lambda |w\rangle$, resultando em

$$(\lambda |v\rangle |w\rangle) \otimes ((|a\rangle + |b\rangle) \otimes |c\rangle)$$

Em seguida, a terceira propriedade pode ser aplicada ao trecho $(|a\rangle + |b\rangle) \otimes |c\rangle$, produzindo o seguinte resultado:

$$(\lambda |v\rangle |w\rangle)(|a\rangle |c\rangle + |b\rangle |c\rangle)$$

Por fim, é possível aplicar a segunda propriedade ao trecho resultante:

$$\lambda |v\rangle |w\rangle |a\rangle |c\rangle + \lambda |v\rangle |w\rangle |b\rangle |c\rangle .$$

A expressão inicial e a expressão obtida após a aplicação das propriedades são equivalentes, resultando no mesmo vetor.

Em muitos casos, a exemplo do processamento de informação, é preciso trabalhar com muitas partículas e o produto tensorial é o ferramental da Mecânica Quântica que permite que isto seja possível.

1.3 Operadores Lineares

Os *operadores lineares* definem *operações* que podem ser aplicadas aos vetores.

Em uma analogia com os números naturais, estas operações equivalem ao conjunto de operações que podem ser aplicadas a estes números, tais como soma, subtração, multiplicação e divisão inteira.

No caso dos vetores, os operadores lineares são *matrizes quadradas*. Seja U um operador linear, $|u\rangle$ e $|v\rangle$ vetores e λ um escalar, U tem as seguintes propriedades:

1. A aplicação do operador U à uma soma de vetores é equivalente à soma da aplicação do operador U a cada vetor:

$$U\left(|u\rangle + |v\rangle\right) = U\left(|u\rangle\right) + U\left(|v\rangle\right)$$

2. A aplicação do operador U ao produto do escalar λ por um vetor é equivalente ao produto do escalar λ pela aplicação do operador U ao vetor:

$$U\left(\lambda \cdot |u\rangle\right) = \lambda \cdot U\left(|u\rangle\right)$$

Exemplo 1.11: Qual o resultado da operação $U \cdot (\lambda \cdot |\psi\rangle + |\varphi\rangle)$, em que U é um operador linear definido pela matriz quadrada $U = \begin{bmatrix} 0 & 3 \\ 3 & 0 \end{bmatrix}$, $|\psi\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ e $|\varphi\rangle = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$ são vetores e $\lambda = 5$?

Sabendo que U é um operador linear, é possível fazer uso das propriedades referentes aos operadores lineares, resolvendo a operação desejada da seguinte forma:

$$\begin{aligned} U(\lambda|\psi\rangle) + |\varphi\rangle &= \begin{bmatrix} 0 & 3 \\ 3 & 0 \end{bmatrix} \left(5 \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right) \\ &= \begin{bmatrix} 0 & 3 \\ 3 & 0 \end{bmatrix} \begin{bmatrix} 7 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 3 \\ 21 \end{bmatrix}. \end{aligned}$$

O resultado da operação é o vetor $\begin{bmatrix} 3 \\ 21 \end{bmatrix}$.

1.4 Produto Externo

Além de produto interno, também se define o *produto externo* entre vetores.

O produto externo de um ket $|\psi\rangle$ com um bra $\langle\varphi|$, denotado por $|\psi\rangle\langle\varphi|$ de acordo com a notação de Dirac, resulta em um operador.

Sejam $|\psi\rangle$ e $|\varphi\rangle$ os seguintes vetores:

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \qquad |\varphi\rangle = \begin{bmatrix} \gamma \\ \delta \end{bmatrix}$$

Para obter o operador resultante do produto externo $|\psi\rangle\langle\varphi|$, basta efetuar o produto das matrizes que representam $|\psi\rangle$ e $\langle\varphi|$, nesta ordem, tal como apresentado a seguir:

$$\begin{aligned} |\psi\rangle\langle\varphi| &= \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \begin{bmatrix} \gamma^* & \delta^* \end{bmatrix} \\ &= \begin{bmatrix} \alpha\gamma^* & \alpha\delta^* \\ \beta\gamma^* & \beta\delta^* \end{bmatrix} \end{aligned}$$

1.5 Autovalores e autovetores

Um *autovetor* de um operador A em um espaço vetorial é um vetor $|v\rangle$ tal que $A|v\rangle = a|v\rangle$, em que a é um número complexo.

A constante a é conhecida como *autovalor* de A associado ao autovetor $|v\rangle$.

Os autovetores e autovalores de um operador são encontrados a partir da *equação característica*. Esta equação é definida como:

$$c(a) \equiv \det|A - a\mathbb{I}|$$

em que $\mathbb{I}$ é a matriz identidade.

As raízes da equação característica $c(a) = 0$ são os autovalores de A.

Exemplo 1.12: Quais os autovalores e autovetores do opera-

dor representado pela matriz identidade $\mathbb{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$?

Para encontrar os autovetores e autovalores do operador $\mathbb{I}$ é necessário encontrar as raízes da equação característica:

$$\begin{aligned} c(a) &= \det|\mathbb{I} - a \cdot \mathbb{I}| \\ &= \det \left| \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right| \\ &= \det \left| \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} -a & 0 \\ 0 & -a \end{bmatrix} \right| \\ &= \det \left| \begin{bmatrix} 1-a & 0 \\ 0 & 1-a \end{bmatrix} \right| \end{aligned}$$

O determinante de uma matriz de ordem 2 é obtido através do produto dos elementos da diagonal principal subtraindo o produto dos elementos da diagonal secundária.

$$\begin{aligned} c(a) &= [(1-a) \cdot (1-a)] - 0 \\ &= a^2 - 2 \cdot a + 1 \end{aligned}$$

Para encontrar os autovalores, basta igualar a equação característica a zero.

$$c(a) = a^2 - 2 \cdot a + 1 = 0$$

A equação característica apresenta duas raízes coincidentes $a_1 = a_2 = 1$.

Para encontrar os autovetores do operador $\mathbb{I}$, basta encontrar os vetores $|v\rangle$ que satisfazem à igualdade $\mathbb{I}|v\rangle = a|v\rangle$. Como existem dois autovalores, existem dois autovetores associados.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = 1 \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}.$$

Para satisfazer a igualdade acima, quaisquer valor de a e b são possíveis (visto que $a = a$ e $b = b$). Sejam escolhidos os pares de valores $a = 3$ e $b = 4$ e $a = 6$ e $b = 5$. No exemplo em questão, a escolha destes valores é quaisquer, ou seja, uma escolha aleatória dos valores de a e b determina autovetores possíveis. Existem casos em que isto não acontece.

Assim, os autovalores do operador $\mathbb{I}$ são $a_1 = a_2 = 1$ e os autovetores associados são $\begin{bmatrix} 3 \\ 4 \end{bmatrix}$ e $\begin{bmatrix} 6 \\ 5 \end{bmatrix}$.

1.6 Operadores de Projeção

Nos casos particulares em que se efetua o produto externo de um vetor com ele mesmo, obtém-se um operador que é denominado *operador de projeção*.

Exemplo 1.13: Qual a representação matricial do operador de projeção obtido a partir do vetor $|\psi\rangle = \begin{bmatrix} 1 \\ i \\ 0 \end{bmatrix}$?

Para obter a representação matricial do operador de projeção desejado basta efetuar o produto externo do vetor $|\psi\rangle$ com o $\langle\psi|$, conforme apresentado a seguir:

$$\begin{aligned} |\psi\rangle\langle\psi| &= \begin{bmatrix} 1 \\ i \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & -i & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 & -i & 0 \\ i & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \end{aligned}$$

O operador de projeção $|\psi\rangle\langle\psi|$ é a matriz de ordem 3 apresentada acima.

Os operadores de projeção quando aplicados a um vetor de um espaço vetorial, projetam este vetor em um *subespaço vetorial* deste espaço. O subespaço vetorial em questão é definido pelos autovalores do vetor que definiu o operador de projeção.

Um *subespaço vetorial* V' é um subconjunto de um espaço vetorial V que satisfaz as seguintes propriedades:

1. O vetor nulo pertence a V';

2. Se dois vetores v e w de V pertencem a V', então $u+w \in V'$;

3. Se $u \in V'$ e a é uma constante tal que $a \in \mathbb{R}$, então $a \cdot v \in V'$.

Exemplo 1.14: Qual a projeção do vetor $|\psi\rangle = \begin{bmatrix} 2 \\ i \end{bmatrix}$ no subespaço vetorial definido pelo autovalores do operador $|\varphi\rangle\langle\varphi|$, em que $|\varphi\rangle = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$?

Para projetar o vetor $|\psi\rangle$ no subespaço vetorial desejado é necessário obter a representação matricial do operador $|\varphi\rangle\langle\varphi|$:

$$\begin{aligned} |\varphi\rangle\langle\varphi| &= \begin{bmatrix} 3 \\ 1 \end{bmatrix} \begin{bmatrix} 3 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 9 & 3 \\ 3 & 1 \end{bmatrix} \end{aligned}$$

A projeção do vetor $|\psi\rangle$ no subespaço vetorial definido pelo operador $|\varphi\rangle\langle\varphi|$ é obtido do seguinte modo:

$$\begin{aligned} |\varphi\rangle\langle\varphi|\,|\psi\rangle &= \begin{bmatrix} 9 & 3 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ i \end{bmatrix} \\ &= \begin{bmatrix} 18+3\cdot i \\ 6+i \end{bmatrix}. \end{aligned}$$

Portanto, a projeção do vetor $|\psi\rangle$ é dada pela seguinte matriz-coluna:

$$\begin{bmatrix} 18+3\cdot i \\ 6+i \end{bmatrix}$$

Uma propriedade importante sobre operadores é a *relação de completude*. A relação de completude estabelece que o somatório de todos os projetores obtidos a partir de vetores de uma base de um espaço vetorial é igual à matriz identidade.

De acordo com a relação de completude, sejam os projetores P_j obtidos a partir dos vetores $|j\rangle$ de uma base B de um espaço vetorial de dimensão j. A seguinte igualdade é verdadeira:

$$\sum_j P_j = \sum_j |j\rangle \langle j| = \mathbb{I}$$

Exemplo 1.15: Seja $B = \{|x\rangle, |y\rangle\}$ uma base para um espaço de Hilbert de dimensão 2, em que $|x\rangle = \begin{bmatrix} i \\ 0 \end{bmatrix}$ e $|y\rangle = \begin{bmatrix} 0 \\ i \end{bmatrix}$. Verifique a relação de completude para os projetores obtidos a partir dos vetores desta base.

Para verificar a relação de completude, é necessário obter os projetores advindos dos vetores da base B. Para tanto, é necessário efetuar o produto externo de cada vetor consigo mesmo.

Para o vetor $|x\rangle$:

$$\begin{aligned} P_x &= \begin{bmatrix} i \\ 0 \end{bmatrix} \cdot \begin{bmatrix} -i & 0 \end{bmatrix} \\ &= \begin{bmatrix} -i^2 & 0 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \end{aligned}$$

De modo análogo, para o vetor $|y\rangle$:

$$\begin{aligned} P_y &= \begin{bmatrix} 0 \\ i \end{bmatrix} \cdot \begin{bmatrix} 0 & -i \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & -i^2 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

Por fim, para verificar a relação de completude, basta mostrar que $P_x + P_y = \mathbb{I}$. Desse modo, tem-se:

$$\begin{aligned} P_x + P_y &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \mathbb{I} \end{aligned}$$

Como a soma de P_x e P_y foi igual à matriz identidade, verifica-se que a relação de completude é satisfeita.

Notas do Capítulo

Neste capítulo foi apresentado o ferramental teórico da Álgebra Linear necessário para dar suporte ao aprendizado dos conceitos da Mecânica Quântica.

Existem outros conceitos necessários e que serão apresentados ao longo desta obra, à medida que forem necessários.

Para aqueles leitores que desejam adquirir mais conhecimentos relacionados à Álgebra Linear necessários para a Computação Quântica, recomenda-se a leitura do capítulo II da obra de Nielsen & Chuang [33] e dos capítulos II, III e IV da obra de McMahon [26].

Exercícios Propostos

1. Seja o vetor $|\varphi\rangle = \begin{bmatrix} \frac{1}{\sqrt{4}} \\ \sqrt{\frac{3}{4}} \end{bmatrix}$.

   (a) Qual é o vetor $\langle\varphi|$?

   (b) Escreva o vetor $|\varphi\rangle$ como uma combinação linear dos vetores da base $\left\{ \begin{bmatrix} i \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ i \end{bmatrix} \right\}$

   (c) Escreva o vetor $|\varphi\rangle$ como uma combinação linear dos

vetores da base de Hadamard B_H dada a seguir:

$$B_H = \{|+\rangle, |-\rangle\}$$

em que

$$|+\rangle = \frac{\begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}}{\sqrt{2}}$$

$$|-\rangle = \frac{\begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \end{bmatrix}}{\sqrt{2}}$$

(d) Qual o valor do produto interno $\langle\varphi|\,\chi\rangle$ sabendo que $|\chi\rangle = \frac{1}{\sqrt{5}}|0\rangle + \sqrt{\frac{4}{5}}|1\rangle$? Qual a relação do valor obtido com o valor $\langle\chi|\,\varphi\rangle$?

(e) Qual a norma do vetor $|\varphi\rangle$?

2. Mostre que o produto interno $\langle\psi|\,\phi\rangle$ dos vetores $|\psi\rangle$ e $|\phi\rangle$ quaisquer é conjugado linear no primeiro argumento:

$$\sum_{i=0}^{n-1} \langle\lambda_i\psi_i|\,\phi\rangle = \sum_{i=0}^{n-1} \lambda_i^* \langle\psi_i|\,\phi\rangle$$

3. Considerando a base de Hadamard, qual o valor do produto interno $\langle+|-\rangle$? Levando em consideração este resultado, o que se pode dizer a respeito destes vetores?

4. Seja o vetor $|\sigma\rangle = \sqrt{\frac{4\cdot i}{7}}|0\rangle + \sqrt{\frac{7-4\cdot i}{7}}|1\rangle$. Quem são os vetores $|\sigma\rangle^{\otimes 2}$, $|\sigma\rangle^{\otimes 3}$ e $|\sigma\rangle^{\otimes 4}$?

5. Dados $\langle a|\, b\rangle = 6$ e $\langle c|\, d\rangle$, calcule o produto interno $\langle \psi|\, \varphi\rangle$ sabendo que $|\psi\rangle = |a\rangle \otimes |c\rangle$ e que $|\varphi\rangle = |b\rangle \otimes |d\rangle$.

6. Denote a representação matricial resultante dos produtos tensoriais a seguir:

   (a) $|+\rangle \otimes |-\rangle$

   (b) $|-\rangle \otimes |+\rangle$

   (c) $|+\rangle^3$

   (d) $\begin{bmatrix} i \\ -1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ -2 \end{bmatrix}$

   (e) $\mathbf{0} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ ($\mathbf{0}$ denota o vetor nulo)

7. Obtenha a representação matricial dos operadores a seguir:

   (a) $|+\rangle \langle -|$

   (b) $|-\rangle \langle +|$

   (c) $|-\rangle \langle -|$

   (d) $|x\rangle \langle x|$ em que $|x\rangle = \begin{bmatrix} 3 \\ -i \\ 2+i \end{bmatrix}$.

   (e) $|+\rangle \langle -| \otimes |-\rangle \langle +|$

8. Determine os autovetores e autovalores da matriz T, cuja representação matricial é dada por:

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i \cdot \pi}{4}} \end{bmatrix}$$

9. Qual o resultado da projeção dos vetores $\begin{bmatrix} i \\ 0 \end{bmatrix}$ e $\begin{bmatrix} 0 \\ i \end{bmatrix}$ nos subespaços vetoriais definidos pelos operadores $P_{|+\rangle}$ e $P_{|-\rangle}$ advindos dos vetores $|+\rangle$ e $|-\rangle$, respectivamente, da base de Hadamard?

10. Mostre que os operadores obtidos a partir dos vetores que compõem a base de Hadamard satisfazem à relação de completude.

Capítulo 2

Noções Gerais da Mecânica Quântica

Este capítulo visa apresentar os conceitos fundamentais para a compreensão da Mecânica Quântica, que por sua vez será fundamental para o aprendizado dos conceitos da Teoria da Computação Quântica. Os conceitos da Mecânica Quântica serão apresentados em função de seus postulados, que descrevem características da Física das partículas.

2.1 Sistemas Quânticos

Apesar das várias descobertas científicas, ainda não se tem uma descrição completa de como o mundo físico realmente funciona. Em virtude disso, um determinado modelo ou teoria pode ser considerado como um instrumento adequado se a diferença entre os resultados esperados e as observações dos fatos estiver abaixo de um certo limiar [15].

Neste contexto, a *Mecânica Quântica* é parte componente da Teoria Quântica e visa dar suporte à uma descrição da natureza, quando se leva em consideração a Física das partículas subatômicas. Para tanto, a Mecânica Quântica possui um conjunto de

postulados, que serão apresentados ao longo deste capítulo.

O primeiro postulado da Mecânica Quântica vincula os conceitos matemáticos apresentados na Seção 1.1 ao estado de um sistema físico. Mais especificamente, o primeiro postulado enuncia que:

Primeiro Postulado da Mecânica Quântica

Qualquer sistema físico isolado pode ser descrito por um vetor unitário em um espaço de Hilbert.

Exemplo 2.1: Um sistema quântico pode ser descrito pelo vetor $|\psi\rangle = \begin{bmatrix} 5 \\ i \end{bmatrix}$? Em caso negativo, obtenha, a partir do vetor $|\psi\rangle$, um vetor capaz de representar um sistema quântico.

Para verificar se o vetor $|\psi\rangle$ pode descrever um sistema físico, é preciso verificar se este é um vetor unitário:

$$\begin{aligned} ||\langle\psi|\,|\psi\rangle|| &= \sqrt{\begin{bmatrix} 5 & -i \end{bmatrix} \begin{bmatrix} 5 \\ i \end{bmatrix}} \\ &= \sqrt{[5^2 - i^2]} \\ &= \sqrt{26} \end{aligned}$$

Como a norma do vetor $|\psi\rangle$ é igual a $\sqrt{26}$, o vetor $|\psi\rangle$ não é unitário e, portanto, não pode ser utilizado para representar um sistema quântico.

Para obter um novo vetor a partir do vetor $|\psi\rangle$ capaz de representar um sistema quântico, basta dividir o vetor $|\psi\rangle$ pela sua norma, obtendo assim um vetor unitário.

$$\begin{aligned} \frac{|\psi\rangle}{||\psi\rangle||} &= \frac{1}{\sqrt{26}} \begin{bmatrix} 5 \\ i \end{bmatrix} \\ &= \begin{bmatrix} \frac{5}{\sqrt{26}} \\ \frac{i}{\sqrt{26}} \end{bmatrix} \end{aligned}$$

Desse modo, o vetor $\frac{|\psi\rangle}{||\psi\rangle||}$ é um vetor unitário adequado para a representação de um sistema quântico.

Os sistemas físicos de dois estados são os de interesse para a Computação Quântica e, em virtude do postulado apresentado, podem ser descritos por um vetor unitário $|\psi\rangle$ em um espaço de Hilbert bidimensional ($\mathbb{C}^2$):

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

em que $\alpha, \beta \in \mathbb{C}$ e, por $|\psi\rangle$ ser um vetor unitário, a restrição $|\alpha|^2 + |\beta|^2 = 1$ deve ser satisfeita.

Qualquer vetor no espaço $\mathbb{C}^2$ pode ser escrito como uma combinação linear dos vetores de uma base deste espaço. Uma

base canônica para o espaço $\mathbb{C}^2$ é o conjunto:

$$B = \{|0\rangle, |1\rangle\}$$

Esta base é conhecida como "*base computacional*". Os kets $|0\rangle$ e $|1\rangle$ possuem a seguinte representação:

$$|0\rangle \equiv \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle \equiv \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Deste modo, no caso do vetor $|\psi\rangle$, este pode ser descrito como a seguinte combinação linear:

$$\begin{aligned} |\psi\rangle &= \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \\ &= \alpha \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= \alpha |0\rangle + \beta |1\rangle \end{aligned}$$

Exemplo 2.2: Denote o vetor $|\psi\rangle = \begin{bmatrix} \frac{i}{\sqrt{3}} \\ \sqrt{\frac{4}{3}} \end{bmatrix}$ como uma combinação linear dos vetores da base computacional.

Para denotar o vetor $|\psi\rangle$ como uma combinação linear dos vetores da base computacional, é preciso saber quais valores de α e de β satisfazem a igualdade:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

Para tanto, basta substituir a representação de kets dos vetores $|\psi\rangle$, $|0\rangle$ e $|1\rangle$ pela representação matricial equivalente e, em seguida, resolver a igualdade para determinar os valores de α e β. Seguindo o procedimento descrito, tem-se:

$$\begin{aligned}\begin{bmatrix} \frac{i}{\sqrt{3}} \\ \sqrt{\frac{4}{3}} \end{bmatrix} &= \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} \alpha \\ \beta \end{bmatrix}\end{aligned}$$

Desse modo, é possível concluir que $\alpha = \frac{i}{\sqrt{3}}$ e que $\beta = -1 + i$. Assim, a combinação linear em termos dos vetores da base computacional pode ser escrita da seguinte forma:

$$|\psi\rangle = \left(\frac{i}{\sqrt{3}}\right)|0\rangle + \left(\sqrt{\frac{4}{3}}\right)|1\rangle$$

Para denotar o estado geral de um sistema quântico de n estados, utiliza-se a seguinte representação:

$$|\psi_n\rangle \equiv \sum_{i=0}^{n-1} \lambda_i\,|i\rangle$$

O estado geral de um sistema quântico de n estados também pode ser representado a partir de um produto tensorial:

$$|\psi_0\rangle \otimes |\psi_1\rangle \otimes \ldots \otimes |\psi_k\rangle$$

Apesar de descrever sistemas quânticos de n estados, o produto tensorial não é capaz de representar *todos* os sistemas quânticos de n estados possíveis.

Os chamados *estados de Bell*, por exemplo, são sistemas quânticos de n estados, mas que não podem ser representados por um produto tensorial. Os estados de Bell são:

$$\begin{aligned}
|b_0\rangle &= \frac{|01\rangle + |10\rangle}{\sqrt{2}} \\
|b_1\rangle &= \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|01\rangle \\
|b_2\rangle &= \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle \\
|b_3\rangle &= \frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|10\rangle
\end{aligned}$$

2.2 Qubits

Na computação clássica, a unidade básica de informação é o *bit* (*binary digit*), que assume valor 0 ou 1 (falso ou verdadeiro, respectivamente). A representação de uma informação é feita por meio da sua codificação em uma seqüência finita de bits [46].

Na Computação Quântica, a unidade básica de representação da informação é um sistema quântico de dois estados: o *qubit* (*quantum bit*).

Um qubit é representado por um vetor unitário em um espaço de Hilbert bidimensional ($|\psi\rangle$):

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{2.1}$$

em que α e $\beta \in \mathbb{C}$. Os valores de α e β devem satisfazer à equação $|\alpha|^2 + |\beta|^2 = 1$, para que o vetor seja unitário. Diz-se que α e β são as *amplitudes* associadas aos kets $|0\rangle$ e $|1\rangle$.

Exemplo 2.3: O vetor $|+\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$ pode representar um qubit em um espaço de Hilbert bidimensional?

Para que o vetor $|+\rangle$ represente um qubit em um espaço de Hilbert bidimensional, duas condições devem ser satisfeitas:

1. O vetor $|+\rangle$ deve pertencer a um espaço de Hilbert bidimensional; e
2. A norma do vetor $|+\rangle$ deve ser unitária.

A primeira condição é satisfeita, pois o vetor $|+\rangle$ é dado pela soma dos vetores $|0\rangle$ e $|1\rangle$ que pertencem a um espaço de Hilbert bidimensional, e a soma é uma operação fechada em um espaço vetorial.

Para verificar se a segunda condição é satisfeita, basta verificar se a norma do vetor $|+\rangle$ é igual a 1 ou, equivalentemente, se a igualdade $|\alpha|^2 + |\beta|^2 = 1$ é satisfeita.

Os valores de α e β do vetor $|+\rangle$ são, ambos, iguais a $\frac{1}{\sqrt{2}}$, pois:

$$|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

Como $\left(\frac{1}{\sqrt{2}}\right)^2 + \left(\frac{1}{\sqrt{2}}\right)^2 = 1$, então $|+\rangle$ é um vetor unitário.

Por satisfazer às duas condições necessárias, então o vetor $|+\rangle$ pode representar um qubit em um espaço de Hilbert bidimensional.

Um qubit pode ser visto como um vetor contido em uma esfera, como ilustra a Figura 2.1.

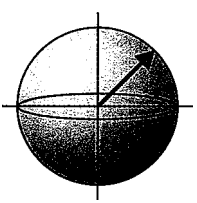


Figura 2.1: Representação gráfica de um qubit [52].

O ângulo que o vetor faz com eixo vertical refere-se às contribuições dos autovetores $|\psi_0\rangle$ e $|\psi_1\rangle$ ao estado do qubit. O ângulo correspondente à rotação em torno do eixo vertical indica a "*fase*".

Segundo a representação de um vetor contido em uma esfera, os qubits $|0\rangle$ e $|1\rangle$ possuem a seguinte representação:

Da mesma forma que a codificação da informação na computação clássica é feita em uma seqüência finita de bits, a re-

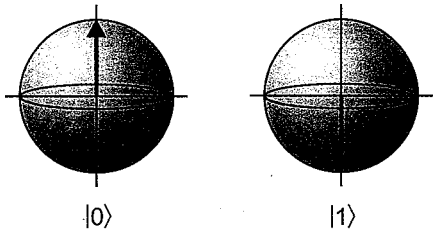


Figura 2.2: *O pólo norte da esfera representa o autovetor* $|0\rangle$ *e o pólo sul representa o autovetor* $|1\rangle$

presentação de uma informação na computação quântica é feita pela codificação da mesma em uma seqüência finita de qubits.

2.3 Superposição

Um qubit pode ser decomposto em estados de uma base. No caso da base computacional, a seguinte decomposição é possível:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

Nos casos em que $\alpha = 0$, tem-se o qubit $|1\rangle$ e nos casos em que $\beta = 1$, tem-se o qubit $|0\rangle$.

Mas nos casos em que os valores de α e β são diferentes de zero simultaneamente $(\alpha, \beta \neq 0)$, diz-se que o qubit está em uma *superposição* destes estados. Quando um qubit está em superposição, não é possível afirmar se este qubit está em $|0\rangle$ ou $|1\rangle$.

Exemplo 2.4: O qubit cujo estado é descrito pelo vetor $|-\rangle$

está em superposição?

Para verificar se o qubit $|-\rangle$ está em superposição, basta verificar se os valores de α e β são diferentes de zero simultaneamente.

$$|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

como $\alpha = \frac{1}{\sqrt{2}}$ e $\beta = \frac{1}{\sqrt{2}}$, ou seja, $\alpha, \beta \neq 0$, diz-se que o qubit $|-\rangle$ está em uma superposição dos estados $|0\rangle$ e $|1\rangle$.

De acordo com a representação de um vetor contido em uma esfera, os exemplos da Figura 2.3 representam superposições.

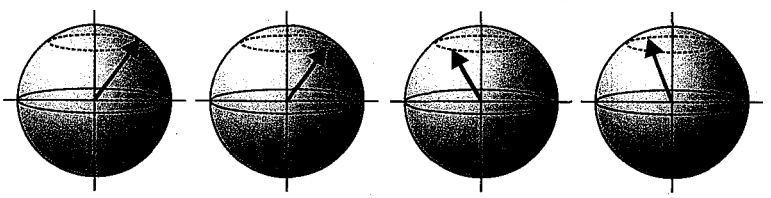


Figura 2.3: *Os qubits representados acima estão em superposição.*

No caso de um qubit que está em superposição e os módulos de suas amplitudes são iguais, ou seja, $|\alpha| = |\beta|$, diz-se que este qubit está em uma *superposição igualmente distribuída* de estados.

Exemplo 2.5: O qubit $|\varphi\rangle = -\frac{|0\rangle}{\sqrt{2}} - \frac{|1\rangle}{\sqrt{2}}$ está em uma super-

posição igualmente distribuída de estados?

Para verificar se o qubit $|\varphi\rangle$ está em uma superposição igualmente distribuída de estados, basta verificar se o módulo de suas amplitudes são iguais.

Tem-se que as amplitudes desse qubit são $\alpha = -\frac{1}{\sqrt{2}}$ e $\beta = -\frac{1}{\sqrt{2}}$. Como $\alpha = \beta$, então $|\alpha| = |\beta|$. Deste modo, o qubit $|\varphi\rangle$ está em uma superposição igualmente distribuída de estados.

2.4 Operadores

Na Computação Clássica, o processamento da informação é realizado pela aplicação de operações aos bits que armazenam a informação. Esses bits são modificados de acordo a operação aplicada. A título de ilustração, a operação lógica *NOT* é a operação mais simples que pode ser aplicada a um bit. Esta operação realiza a inversão do bit de entrada, como descrito na Tabela 2.1.

Tabela 2.1: *Operação Lógica NOT*

Entrada	Saída
0	1
1	0

Na Computação Quântica o processamento da informação

também é realizado através de operadores, cuja definição está associada ao segundo postulado da Mecânica Quântica. Esse postulado relaciona a aplicação de operadores à evolução de sistemas quânticos, enunciando que:

Segundo Postulado da Mecânica Quântica

Um sistema quântico isolado originalmente no estado $|\psi_1\rangle$ evolui para o estado $|\psi_2\rangle$ por meio da aplicação de um operador unitário U:

$$|\psi_2\rangle = U\,|\psi_1\rangle$$

Um operador unitário U é um operador que tem a seguinte propriedade:

$$U^\dagger \cdot U = U \cdot U^\dagger = \mathbb{I}$$

em que $U^\dagger = (U^*)^T$, ou seja, a conjugada transposta de U e $\mathbb{I}$ é a matriz identidade. $U^\dagger$ é o operador inverso do operador U, ou seja, $U^\dagger$ equivale a U^{-1}.

Exemplo 2.6: Seja o operador $E = \begin{bmatrix} -i & 0 \\ 0 & 1 \end{bmatrix}$. Verifique se E é um operador unitário.

Para verificar se o operador E é unitário, basta verificar se este satisfaz à equação $E^\dagger E = \mathbb{I}$.

Para tanto, é necessária a representação de $E^\dagger$:

$$\begin{aligned} E^\dagger &= \left(\begin{bmatrix} -i & 0 \\ 0 & 1 \end{bmatrix}^* \right)^T \\ &= \begin{bmatrix} -i^* & 0^* \\ 0^* & 1^* \end{bmatrix}^T \\ &= \begin{bmatrix} i & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

O próximo passo é verificar o resultado do produto $E^\dagger E$:

$$\begin{aligned} E \cdot E^\dagger &= \begin{bmatrix} -i & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} i & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} -i^2 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \mathbb{I} \end{aligned}$$

Assim, como $E^\dagger E = \mathbb{I}$, é possível afirmar que o operador E é um operador unitário.

Por serem unitários, os operadores na Computação Quântica também são *operadores lineares*. Vários operadores unitários são definidos na Computação Quântica. Dentre eles, destacam-se as *matrizes de Pauli*:

$$\mathbb{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

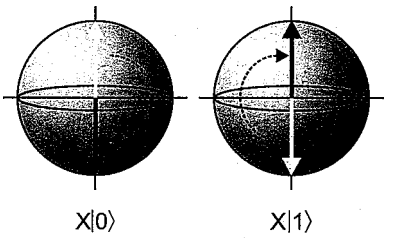
$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

O operador X, em particular, merece destaque, pois é o análogo quântico da porta NOT clássica. Quando o operador X é aplicado ao qubit $|1\rangle$, produz o seguinte resultado:

$$\begin{aligned} X|1\rangle &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ &= |0\rangle \end{aligned}$$

Observando os resultados da aplicação do operador X aos qubits $|0\rangle$ e $|1\rangle$, é interessante notar que o efeito da aplicação deste operador é equivalente à inversão do qubit ao qual ele se aplica, ou seja, equivale à porta NOT da Computação Clássica, que realiza a inversão dos bits de entrada.

Utilizando a representação de um vetor contido em uma fera, a aplicação da porta X pode ser visualizada conforme strado na Figura 2.4.



gura 2.4: *Representação da aplicação do operador X aos qubits $|0\rangle$ e $|1\rangle$. vetor na cor cinza representa o estado inicial e a seta tracejada indica sentido da rotação.*

Outro operador importante para a Computação Quântica rque sua aplicação é capaz de produzir superposições igual-ente distribuídas é o *operador de Hadamard*, denotado por H ja representação matricial é dada a seguir:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

O operador de Hadamard aplicado aos qubits $|0\rangle$ e $|1\rangle$, por emplo, cria as seguintes superposições:

$$\begin{aligned} H|0\rangle &= \frac{1}{\sqrt{2}} \cdot |0\rangle + \frac{1}{\sqrt{2}} \cdot |1\rangle = |+\rangle \\ H|1\rangle &= \frac{1}{\sqrt{2}} \cdot |0\rangle - \frac{1}{\sqrt{2}} \cdot |1\rangle = |-\rangle \end{aligned}$$

Exemplo 2.7: Qual o estado final de um sistema quântico $|$
após a operação $X|-\rangle$?

É interessante perceber que o estado $|-\rangle = \frac{|0\rangle-|1\rangle}{\sqrt{2}}$ é un
superposição e para que a este estado seja aplicado o operad
X, é necessário utilizar as propriedades dos operadores lineare

$$\begin{aligned} X\left(\frac{1}{\sqrt{2}}\cdot|0\rangle - \frac{1}{\sqrt{2}}\cdot|1\rangle\right) &= X\left(\frac{1}{\sqrt{2}}\cdot|0\rangle\right) - X\left(\frac{1}{\sqrt{2}}\cdot|1\rangle\right) \\ &= \frac{1}{\sqrt{2}}X|0\rangle - \frac{1}{\sqrt{2}}X|1\rangle \\ &= \frac{1}{\sqrt{2}}|1\rangle - \frac{1}{\sqrt{2}}|0\rangle \end{aligned}$$

Assim, o estado final do sistema quântico em questão é un
superposição de estados denotada por $\frac{|1\rangle-|0\rangle}{\sqrt{2}}$.

2.4.1 Produto Tensorial de Operadores

Para que operadores quânticos possam ser aplicados à produt
tensoriais de estados, é necessário que a definição dos mesm
dê suporte à este tipo de produto. De um modo mais simp
ficado, é preciso que o produto tensorial possa ser aplicado
operadores.

Sejam então A um operador quântico de dimensões $m \times n$
B outro operador cujas dimensões são $p \times q$. O produto $A \otimes$
resulta em:

$$A \otimes B \equiv \begin{bmatrix} A_{11}B & A_{12}B & \dots & A_{1n}B \\ A_{21}B & A_{22}B & \dots & A_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1}B & A_{m2}B & \dots & A_{mn}B \end{bmatrix}$$

O operador resultante, $A \otimes B$, possui dimensões $n \cdot q \times m \cdot p$ e pode então ser aplicado à um produto tensorial de vetores de modo já conhecido.

Exemplo 2.8: Qual o resultado da aplicação do operador $X^{\otimes 2}$ ao estado $|0\rangle^{\otimes 2}$?

O primeiro passo é obter a representação matricial do operador $X^{\otimes 2}$.

$$X \otimes X = X^{\otimes 2} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

É interessante observar que simplesmente o operador X não poderia ser aplicado, pois este é representado por uma matriz de dimensões 2×2 enquanto que o estado apresentado possui dimensões 4×1. Como a aplicação do operador é uma multiplicação de matrizes, as dimensões das matrizes tornaria a multiplicação incompatível.

O estado $|0\rangle^{\otimes 2}$ é representado pelo vetor:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

O passo seguinte é aplicar o operador $X^{\otimes 2}$ ao estado $|0\rangle^{\otimes 2}$:

$$\begin{aligned} X^{\otimes 2}|0\rangle^{\otimes 2} &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \\ &= |1\rangle \otimes |1\rangle . \end{aligned}$$

O estado resultante da aplicação do operador $X^{\otimes 2}$ ao estado $|0\rangle^{\otimes 2}$ é o estado $|1\rangle^{\otimes 2}$.

2.5 Reversibilidade

Uma conseqüência imediata da definição do processo evolutivo de um sistema quântico como uma operação unitária é a *reversibilidade*. A reversibilidade é uma propriedade que permite

obter o estado inicial de um qubit, dado que se conhece apenas o estado resultante de um conjunto de operações que foram aplicadas neste qubit [52].

De acordo com o segundo postulado da Mecânica Quântica, a reversibilidade permite que o estado original de um sistema pode ser obtido da seguinte forma:

$$|\psi_1\rangle = U^\dagger |\psi_2\rangle$$

Exemplo 2.9: Sabe-se que um qubit é descrito pelo estado $|\varphi\rangle = |1\rangle$ e que este qubit evoluiu a partir da aplicação do operador Z. Qual o estado original desse qubit?

De acordo com a definição de reversibilidade, o qubit original $|\psi_1\rangle$ pode ser obtido da seguinte forma:

$$\begin{aligned}
|\psi_1\rangle &= Z^\dagger |1\rangle \\
&= \left(\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \right)^\dagger \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ -1 \end{bmatrix} = -1 \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= -1 \cdot |1\rangle
\end{aligned}$$

Devido à reversibilidade, é possível concluir que o estado inicial do qubit era $|\varphi\rangle = -1 \cdot |1\rangle$.

2.6 Medição

Um sistema quântico isolado possui sua evolução descrita por transformações unitárias. Mas, para acessar o estado de um sistema quântico é necessário realizar uma tarefa denominada *medição*.

A medição é uma "*interface*" entre os níveis clássico e quântico e é descrita pelo terceiro postulado da Mecânica Quântica:

Terceiro Postulado da Mecânica Quântica

A medição é descrita por operadores denominados "*operadores de medição*", $\{M_m\}$. Estes operadores atuam sobre o espaço de estados do sistema quântico. O índice m se refere aos possíveis resultados. Se o estado de um sistema quântico for $|\psi\rangle$, imediatamente antes da medição, então a probabilidade $p(m)$ do valor m ocorrer como resultado da aplicação dos operadores de medição é dada por:

$$p(m) = \langle\psi| M_m^\dagger M_m |\psi\rangle$$

e o estado $|\psi'\rangle$ do sistema após a medição será

$$|\psi'\rangle = \frac{M_m |\psi\rangle}{\sqrt{\langle\psi| M_m^\dagger M_m |\psi\rangle}}$$

A medição de interesse no âmbito deste livro é a *medição projetiva*. Neste tipo de medição, utiliza-se como conjunto de operadores de medida, um conjunto de projetores obtidos de uma base de um espaço de Hilbert, a exemplo da base computacional. Conforme visto na Seção 1.6, os projetores que provém de uma base de um espaço vetorial obedecem à relação de completude.

Seja um qubit $|\varphi\rangle$ representado por:

Tabela 2.2: Efeitos da medição em um qubit

Estado Inicial	Medição		
	Valor Medido	Estado Final	Probabilidade Associada
$\|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$	0	$\|0\rangle$	$\|\alpha\|^2$
	1	$\|1\rangle$	$\|\beta\|^2$

$$|\varphi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

e um conjunto de operadores de medida de projetores obtidos a partir da base computacional, ou seja, o conjunto $\{|0\rangle\langle 0|, |1\rangle\langle 1|\}$.

O terceiro postulado prediz que uma medição em $|\psi\rangle$ fará com que este qubit assuma o estado $|0\rangle$ com uma probabilidade $|\alpha|^2$ ou o estado $|1\rangle$ com probabilidade $|\beta|^2$. Os efeitos da medição estão descritos na Tabela 2.2.

Para um sistema quântico de n estados, descrito por $\sum_{i=0}^{n-1} \lambda_i |i\rangle$, o raciocínio é análogo: uma medição neste sistema quântico irá gerar como resultado o valor i com uma probabilidade $|\lambda_i|^2$ e fará com que o sistema quântico assuma o estado $|i\rangle$.

Exemplo 2.10: Seja um qubit no estado $|\psi\rangle = |0\rangle$. Considerando como operadores de medida os projetores obtidos a

partir da base computacional, quais os possíveis resultados de uma medição neste qubit?

Os resultados de uma medição são estados que o sistema quântico pode assumir. Como os projetores utilizados são da base computacional, então o sistema quântico poderá assumir os estados $|0\rangle$ ou $|1\rangle$, porém com uma probabilidade associada.

No caso do sistema assumir o estado $|0\rangle$, a probabilidade associada é:

$$\begin{aligned} p(0) &= \langle\psi|\,|0\rangle\,\langle 0|^{\dagger}\,|0\rangle\,\langle 0|\,|\psi\rangle \\ &= \langle 0|\,|0\rangle\,\langle 0|\,|0\rangle\,\langle 0|\,|0\rangle \\ &= 1 \cdot 1 \cdot 1 \\ &= 100\% \end{aligned}$$

De modo análogo, a probabilidade de assumir o estado $|1\rangle$ é dada por:

$$\begin{aligned} p(1) &= \langle\psi|\,|1\rangle\,\langle 1|^{\dagger}\,|1\rangle\,\langle 1|\,|\psi\rangle \\ &= \langle 0|\,|1\rangle\,\langle 1|\,|1\rangle\,\langle 1|\,|0\rangle \\ &= 0 \cdot 1 \cdot 0 \\ &= 0 \end{aligned}$$

Ou seja, uma medição no estado $|\varphi\rangle$ resulta no valor 0, indicando que o qubit assumiu o estado $|0\rangle$, em 100% das vezes.

A figura 2.5 exemplifica o que acontece na medição. Qubits em superposição assumirão os estados $|0\rangle$ ou $|1\rangle$ de acordo como descrito no terceiro postulado.

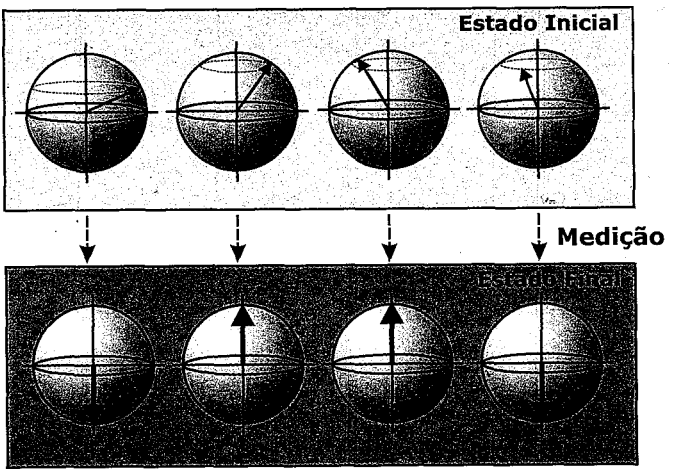


Figura 2.5: *Ilustração do efeito de uma medição em qubits.*

Exemplo 2.11: Qual a probabilidade do qubit $|\varphi\rangle = \frac{1}{\sqrt{3}}|0\rangle + \sqrt{\frac{2}{3}}|1\rangle$ assumir o estado $|0\rangle$ após uma medição projetiva na base computacional?

A probabilidade do qubit $|\varphi\rangle$ assumir o estado $|0\rangle$ pode ser obtida como se segue, de acordo com o terceiro postulado da Mecânica Quântica:

$$
\begin{aligned}
p(0) &= \langle\varphi|\,|0\rangle\,\langle 0|^{\dagger}\,|0\rangle\,\langle 0|\,|\varphi\rangle \\
&= \langle\varphi|\,|0\rangle\,\langle 0|\,|0\rangle\,\langle 0|\,|\varphi\rangle \\
&= \langle\varphi|\,|0\rangle\,\langle 0|\,|\varphi\rangle \\
&= \begin{bmatrix} \frac{1}{\sqrt{3}} & \sqrt{\frac{2}{3}} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{bmatrix} \\
&= \frac{1}{3}
\end{aligned}
$$

Assim, a probabilidade de uma medição no qubit $|\varphi\rangle$ resultar em $|0\rangle$ é igual a $\frac{1}{3}$.

Notas do Capítulo

O capítulo apresentado introduz os conceitos fundamentais da Mecânica Quântica. O aprendizado desses conceitos é essencial para os interessados em qualquer tópico da Computação Quântica ou até mesmo da Física Quântica.

Como outras fontes de estudo para ampliar o conhecimento, sugere-se a leitura do capítulo III da obra de Kaye [18] e dos capítulos II e III da obra de Imre & Balazs [15], em que a Mecânica Quântica é vista em maior profundidade.

Exercícios Propostos

1. Seja um sistema quântico no estado

$$|\psi\rangle = \frac{(1-i)}{\sqrt{3}} \cdot |0\rangle + \frac{1}{\sqrt{3}} \cdot |1\rangle$$

Se uma medição é efetuada, qual a probabilidade do estado do sistema assumir o estado $|0\rangle$? E qual a probabilidade de assumir o estado $|1\rangle$?

2. Encontre $X \otimes Y\,|\varphi\rangle$, em que:

$$|\varphi\rangle = \frac{|0\rangle\,|1\rangle - |1\rangle\,|0\rangle}{\sqrt{2}}$$

3. Um sistema quântico de dois qubits encontra-se no estado $|\phi\rangle$

$$|\phi\rangle = \frac{1}{\sqrt{6}} \cdot |01\rangle + \sqrt{\frac{5}{6}} \cdot |10\rangle$$

Este estado encontra-se normalizado? Um operador X é aplicado ao segundo qubit. Após esta operação, se os dois bits são medidos, quais as probabilidades dos resultados das medições?

4. Considere a base de Hadamard, denotada por B:

$$\begin{aligned} B &= \{|+\rangle, |-\rangle\} \\ &= \left\{ \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \frac{|0\rangle - |1\rangle}{\sqrt{2}} \right\} \end{aligned}$$

(a) Denote a representação da operador de Pauli X relativa a base B.

(b) Mostre que $X = |0\rangle\langle 1| + |1\rangle\langle 0| = P_+ - P_-$.

5. Se

$$|\varphi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

encontre $\mathbb{I} \otimes Y\,|\varphi\rangle$.

Capítulo 3

Teoria dos Autômatos Finitos

A Teoria dos Autômatos é um ramo da Teoria da Computação que estuda modelos computacionais abstratos através de descrições matemáticas. Seu estudo proporciona o entendimento do que é computação e quais os seus limites, sendo de fundamental importância para os cursos de Ciência da Computação.

A seção a seguir[1] apresenta um breve histórico acerca desta teoria, possibilitando que o leitor tome conhecimento da sua importância e de algumas de suas aplicações.

3.1 Breve Histórico

Os principais conceitos da Teoria dos Autômatos foram introduzidos na década de 50 como resultado do esforço de diversos pesquisadores incluindo matemáticos, lingüistas, neurofisiologistas e engenheiros eletricistas. Os trabalhos iniciais da área buscavam o desenvolvimento de máquinas que modelassem os processos cognitivos do cérebro humano. O matemático Alan Turing apresentou, em 1936, a Máquina de Turing como sendo

[1]Extraída da dissertação de Isidro [16]

um modelo de uma máquina cujo funcionamento era baseado no procedimento usado pelos matemáticos para prova de teoremas [48].

Alguns anos mais tarde, em 1943, McCulloch, psiquiatra, e Pitts, matemático, construíram um modelo para explicar o funcionamento dos neurônios utilizando o conceito de máquinas de estado finito [25]. Em 1951, Kleene fez um estudo minucioso sobre o artigo de McCulloch e Pitts, mas somente em 1956, publicou seu trabalho no qual introduziu o conceito de linguagens regulares, originalmente denominadas de "eventos regulares" [20]. Huffman, em 1954, estudando a síntese de circuitos seqüenciais, apresentou a noção de estado de um autômato e de tabela de transição [14]. Moore [32], em 1956, introduziu a noção de estados indistinguíveis e apresentou um algoritmo de minimização, além de desenvolver um modelo de autômato com saída, simultaneamente ao trabalho de Mealy [27]. O conceito de autômatos finitos não-determinísticos foi introduzido por Rabin e Scott [39] que expuseram os principais conceitos da área de forma sistemática, servindo de base para diversos trabalhos posteriores [23]. Para uma revisão histórica mais detalhada sugere-se o artigo de Perrin [36].

Nas décadas de 60 e 70, os estudos teóricos na área foram bastante ativos. Nas últimas duas décadas, no entanto, o foco das pesquisas tem sido as inúmeras aplicações práticas dos conceitos e idéias da Teoria dos Autômatos que envolvem quase todas as áreas da Ciência da Computação [13], dentre as quais

destacam-se:

1. Softwares verificadores de circuitos digitais [37];
2. Analisadores léxicos dos compiladores [1];
3. Diversas aplicações em processamento de linguagem natural, por exemplo, dicionários multilíngüe, thesauri e verificadores de escrita, atuando na representação de vocabulários e indexação de textos [40, 17, 19, 6, 30];
4. Módulos de casamento de padrões utilizados em diversos softwares de busca [11];
5. Geradores de seqüências de números [51];
6. Descrições de algoritmos na teoria dos grupos [43];
7. Processadores de linguagem XML [42];
8. Especificações conceituais de e-Services [7];
9. Verificadores de programas [49];
10. Aplicações em aprendizagem de máquina [41], etc.

Em virtude desse grande alcance prático, a Teoria dos Autômatos deixou de ser útil apenas em sala de aula, como componente teórico dos cursos de Computação, e tornou-se fundamental para o desenvolvimento de aplicações do mundo real.

O *Autômato Finito* (AF) é um dos modelos de computação mais simples, porém com vasta aplicação: processadores de

texto, compiladores, projetos de hardware, etc. Além da simplicidade e do interesse prático, o estudo deste modelo de computação possibilita a prática de definições formais de computação, proporciona o aprendizado de conceitos relevantes e favorece o aprendizado de modelos computacionais mais complexos.

Existem alguns modelos de Autômatos Finitos consagrados pela literatura e que serão abordadas neste livro, são eles: determinístico, não-determinístico, probabilístico e quântico. Antes da apresentação destes modelos, será feita uma breve introdução acerca de conceitos de linguagem, essencial para o aprendizado da Teoria dos Autômatos.

3.2 Linguagem

Entende-se por *alfabeto* um conjunto finito de símbolos, denotado pela letra grega Σ. Alguns alfabetos comumente utilizados são:

1. $\Sigma = \{0, 1\}$ – o alfabeto binário;
2. $\Sigma = \{a, b, c, \ldots, x, y, z\}$ – o alfabeto das letras minúsculas;
3. $\Sigma = \{a, b, \ldots, y, z, 0, 1, \ldots, 8, 9\}$ – o alfabeto alfa-numérico.

Uma *palavra* é qualquer seqüência finita de símbolos de um alfabeto. O *comprimento* de uma palavra ω é a quantidade de símbolos que esta palavra possui e é denotado por $|\omega|$. O i-ésimo símbolo da palavra ω é denotado por $\omega_{(i)}$.

Exemplo 3.1: Quantas palavras de comprimento igual a 3 podem ser obtidas do alfabeto $\Sigma = \{0, 1\}$?

Todas as palavras de comprimento igual a 3 correspondem à todas as combinações possíveis envolvendo três símbolos do alfabeto, ou seja:

$$\omega_1 = 000 \quad \omega_2 = 001$$
$$\omega_3 = 010 \quad \omega_4 = 011$$
$$\omega_5 = 100 \quad \omega_6 = 101$$
$$\omega_7 = 110 \quad \omega_8 = 111.$$

Como são possíveis 8 combinações distintas, é possível obter 8 palavras de comprimento 3.

A *palavra vazia*, denotada por λ, corresponde à palavra com zero ocorrências de símbolos, ou seja, $|\lambda| = 0$.

Duas operações básicas são definidas sobre palavras:

1. Concatenação: Sejam duas palavras α e β, com comprimentos $|\alpha| = m$ e $|\beta| = n$. A concatenação de α e β, denotada por $\alpha \cdot \beta$, é dada por:

$$\alpha \cdot \beta = \alpha_{(1)} \ldots \alpha_{(m)} \beta_{(1)} \ldots \beta_{(n)}$$

2. Reverso: Seja uma palavra α, o reverso de α, denotado

por α^R, é dado por:

$$\begin{aligned} \lambda^R &= \lambda \\ (\alpha\gamma)^R &= \gamma\alpha^R \end{aligned}$$

em que γ denota um símbolo.

Exemplo 3.2: Sejam as palavras $\alpha = 010$, $\beta = 01$ e a palavra vazia λ. Obtenha as palavras resultantes das operações indicadas.

As palavras resultantes são:

a. $\alpha \cdot \beta = 01001$;

b. $(\alpha \cdot \beta)^R = (01001)^R = 10010$;

c. $\alpha^R \cdot \beta^R = (010)^R \cdot 01^R = 010 \cdot 10 = 01010$

d. $\alpha \cdot \lambda = \alpha$;

e. $\lambda \cdot \lambda = \lambda$.

Considera-se que Σ^* é o conjunto de todas as palavras de Σ, incluindo a palavra vazia. Para qualquer alfabeto Σ, Σ^* é um conjunto infinito enumerável.

Uma *linguagem* sobre um alfabeto Σ é qualquer subconjunto de Σ^*. Dado que linguagens são conjuntos, as operações de união, subtração, intersecção e complemento são aplicáveis.

Exemplo 3.3: Considerando o alfabeto binário e as linguagens $L_1 = \{0\}^* - \{\lambda\}$ e $L2 = \{1\}^* - \{\lambda\}$, qual a linguagem formada pela concatenação $L_1 \cdot L_2$?

A linguagem L_1 é formada por todas as palavras compostas somente por símbolos 0. A linguagem L_2 é composta por todos as palavras compostas somente por símbolos 1.

A concatenação destas linguagens, denotada por $L_1 \cdot L_2$, é uma linguagem composta por palavras da forma $\omega_1 \cdot \omega_2$, em que $\omega_1 \in L_1$ e $\omega_2 \in L_2$. Exemplos de palavras pertencentes à $L_1 \cdot L_2$ são:

$$\{01, 0011, 000011, 011111, \ldots\}$$

Uma característica importante da linguagem $L_1 \cdot L_2$ é que todas as suas palavras iniciam com uma seqüência de zeros seguida por uma seqüência de uns.

3.3 Autômatos Finitos Determinísticos

Um *Autômato Finito* (AF) é um modelo de computação que pode ser pensado como um discriminador de palavras de Σ^*:

palavras que o autômato "aceita" e palavras que o autômato "rejeita". O conjunto das palavras aceitas unido com o conjunto das palavras rejeitadas constitui o conjunto de todas as palavras, ou seja, Σ^*. O conjunto das palavras que o autômato aceita constitui a "linguagem" deste autômato.

O autômato mais simples a realizar este processo é o *Autômato Finito Determinístico* (AFD), que possui a seguinte definição formal:

Definição Formal – Autômato Finito Determinístico

Um Autômato Finito Determinístico é uma 5-tupla $\langle Q, \Sigma, \delta, q_0, F \rangle$, em que:

1. Q é um conjunto finito denominado *conjunto de estados*;
2. Σ é um *alfabeto*;
3. $\delta : Q \times \Sigma \rightarrow Q$ é uma *função de transição*;
4. $q_0 \in Q$ é o *estado inicial*; e
5. $F \subseteq Q$ é o conjunto de *estados de aceitação.*

A função de transição descreve o comportamento do autômato. À medida que uma palavra é lida, símbolo a símbolo, o autômato muda de estado, de acordo com a função de transição

δ: o próximo estado depende do estado atual e do símbolo lido.

Assim, por exemplo:

$$\delta(q_a, x) = q_b$$

indica que se o autômato estiver no estado q_a e for efetuada a leitura do símbolo x, o autômato deve passar para o estado q_b.

É comum a utilização de uma tabela de transição de estados para representar a função de transição de um autômato, como apresentado na Tabela 3.1. Nesta tabela, a primeira coluna representa o estado atual do autômato e a primeira linha representa o símbolo lido. Supondo que este autômato esteja no estado q_1 e tenha lido o símbolo 0, irá assumir o estado q_0.

Tabela 3.1: *Exemplo de uma tabela de transição de estados de um AFD.*

δ	0	1
q_0	q_0	q_1
q_1	q_0	q_1

Para aceitar ou rejeitar uma palavra ω, um autômato finito determinístico procede da seguinte maneira: no momento inicial o autômato está no estado inicial q_0 e tem como entrada a palavra ω. O autômato lê o primeiro símbolo da palavra ($\omega_{(1)}$) e muda de estado (ou permanece no mesmo), de acordo com sua função de transição. Este processo é repetido até que todos os símbolos de ω tenham sido lidos pelo autômato. Uma vez que o estado que o autômato assume após a leitura de cada símbolo

é perfeitamente determinado, este tipo de autômato é nomeado *determinístico*.

Quando o final da palavra é atingido, se o estado atual q pertencer ao conjunto dos estado de aceitação (ou seja, $q \in F$), a palavra é dita ser aceita pelo autômato. Em caso contrário (ou seja, $q \notin F$), a palavra é dita ser rejeitada pelo autômato. O processo de aceitar ou rejeitar uma palavra é conhecido como *computação* de uma palavra por um autômato. O processo de aceitar ou rejeitar uma palavra é conhecido como *computação* de uma palavra por um autômato.

Exemplo 3.4: Verifique se a linguagem $L = \{a\}^*$ é uma linguagem regular.

Para verificar se a linguagem L é regular, basta verificar a existência de um autômato finito determinístico que aceite todas as palavras desta linguagem.

Um AFD A com definição formal dada por $\langle Q, \Sigma, \delta, q_0, F\rangle$ em que $Q = \{q_0\}$, $\Sigma = \{a\}$, $\delta(q_0, a) = q_0$ e $F = \{q_0\}$ é capaz de aceitar todas as palavras da linguagem L.

Portanto, como existe um autômato A que reconhece todas as palavras de L, L é uma linguagem regular.

É conveniente se utilizar uma extensão da função de transição:

$$\hat{\delta} : Q \times \Sigma^* \rightarrow Q$$

definida da seguinte maneira:

$$\begin{aligned} \hat{\delta}(q, \lambda) &= q \\ \hat{\delta}(q, \alpha) &= \delta(q, \alpha), \alpha \in \Sigma \\ \hat{\delta}(q, \alpha\omega) &= \hat{\delta}(\delta(q, \alpha), \omega), \alpha \in \Sigma, \omega \in \Sigma^* \end{aligned}$$

Assim, pode-se definir a linguagem reconhecida por um AFD A como:

$$L(A) = \left\{ \omega \in \Sigma^* : \hat{\delta}(q_0, \omega) \in F \right\}$$

O conjunto de todas as linguagens aceitas por AFD's compõe a classe das *linguagens regulares*.

3.3.1 Diagrama de Estados

Uma forma bastante intuitiva de compreender o funcionamento dos autômatos é através da representação por *diagrama de estados*.

Seja um autômato $A = \langle Q, \Sigma, \delta, q_0, F \rangle$. O diagrama de estados de um autômato A é um *grafo direcionado* que pode ser construído da seguinte forma:

1. Cada estado $q \in Q$ é um nó do grafo;

2. Para os estados $p, q \in Q$ e para cada símbolo $a \in \Sigma$, se existe a transição $\delta(p, a) = q$, então deve existir um arco direcionado de p para q rotulado com o símbolo a;

3. O nó correspondente ao estado inicial deve possuir uma seta; e

4. Os estados de aceitação devem ser marcados com um círculo duplo.

A representação da computação de uma palavra por um diagrama de estados de um AFD pode ser vista como um percurso que tem início no estado inicial e que consome um símbolo da palavra sempre que uma transição é feita. Após a leitura de todos os símbolos da palavra, se o percurso estiver em um dos estados aceitação, significa que o autômato aceita a palavra, em caso contrário, este autômato a rejeita.

Exemplo 3.5: Seja o autômato $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, em que $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{0, 1\}$, $F = \{q_1\}$ e $\delta(q_0, 0) = q_2$, $\delta(q_0, 1) = q_1$, $\delta(q_1, 0) = q_2$, $\delta(q_1, 1) = q_1$. Construa o diagrama de estados relativo a este autômato finito determinístico.

Seguindo os passos para construção do diagrama de estados de um autômato, o autômato A está representado na Figura 3.1.

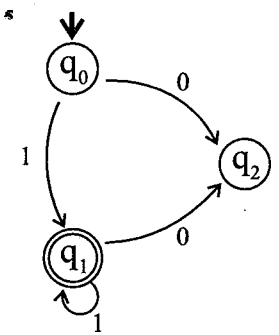


Figura 3.1: *Diagrama de estados do autômato A.*

Observe que o número de estados e as transições no diagrama de estados respeitam a definição do autômato apresentada no enunciado.

3.3.2 Notação Matricial

A *notação matricial* é uma forma simplificada de representar os passos da computação de uma palavra por um autômato. Para representar um autômato $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ segundo a notação matricial, os seguintes passos são necessários:

1. Deve-se ordenar os estados do autômato;

2. Construir o vetor-coluna π, identificando o estado inicial do autômato. O elemento correspondente ao estado inicial deve possuir o valor 1, e os demais elementos devem ser nulos;

3. Construir o vetor-coluna η, identificando os estados de

aceitação do autômato. Os elementos que denotem estados de aceitação devem ser iguais a 1 e os demais iguais a 0;

4. Para cada símbolo $a \in \Sigma$, definir a matriz de transição X_a, na qual as linhas e colunas correspondem aos estados do autômato. A entrada para a linha correspondente ao estado q, e para a coluna correspondente ao estado q' é igual a 1 se $\delta(q, a) = q'$, e 0 caso contrario.

Para verificar se uma palavra ω é aceita pelo autômato A (ou seja, se $\omega \in L(A)$) basta utilizar a seguinte expressão:

$$\pi^T \cdot X_\omega \cdot \eta = \begin{cases} 1, & \text{se } \omega \in L(A) \\ 0, & \text{em caso contrário} \end{cases}$$

em que X_ω é a matriz resultante da multiplicação das matrizes correspondentes aos símbolos da palavra ω.

Exemplo 3.6: Seja o autômato $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, em que $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{0, 1\}$, $F = \{q_1\}$ e $\delta(q_0, 0) = q_2, \delta(q_0, 1) = q_1, \delta(q_1, 0) = q_2, \delta(q_1, 1) = q_1$. Utilizando a notação matricial, verifique se a palavra $\omega = 110$ pertence à linguagem deste autômato.

Para verificar se a palavra ω pertence à linguagem do autômato A, é necessário construir as matrizes π, X_{110} e η para ilustrar a computação através de matrizes de transição.

Considerando a seguinte ordenação dos estados (q_0, q_1, q_2) e sabendo que q_0 é o estado inicial, o vetor-coluna π, que representa o estado inicial, é dado por:

$$\pi = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

Considerando a ordenação definida, o vetor-coluna η dos estados de aceitação é dado por:

$$\eta = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

As matrizes relativas aos símbolos de Σ que indicam as transições dos estados do autômato são:

$$X_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad X_0 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Munidos das matrizes que representam as transições, dado que a palavra $\omega = 110$, basta efetuar a multiplicação $X_1 \cdot X_1 \cdot X_0$, que resulta em:

$$X_{110} = X_1 \cdot X_1 \cdot X_0 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Finalmente, para verificar se a palavra pertence à linguagem do autômato, basta efetuar o produto:

$$\begin{aligned} ACC_\omega &= \pi^T \cdot X_{110} \cdot \eta \\ &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ &= 0. \end{aligned}$$

Como o resultado da expressão foi igual à zero, isto indica que a palavra ω não pertence à $L(A)$.

3.4 Autômatos Finitos não-Determinísticos

Diferentemente dos AFD, os *autômatos finitos não-determinísticos* (AFND) podem assumir um conjunto de estados em cada passo da computação. Esta diferença é ilustrada na Figura 3.2 por meio da comparação das computações nestes dois tipos de autômatos.

A computação de uma palavra ω por um AFND se dá de modo similar a um AFD, diferenciando-se apenas pela possibilidade do autômato assumir mais de um estado ao mesmo tempo. Em conseqüência desta diferença, a definição formal dos AFND's é dada por:

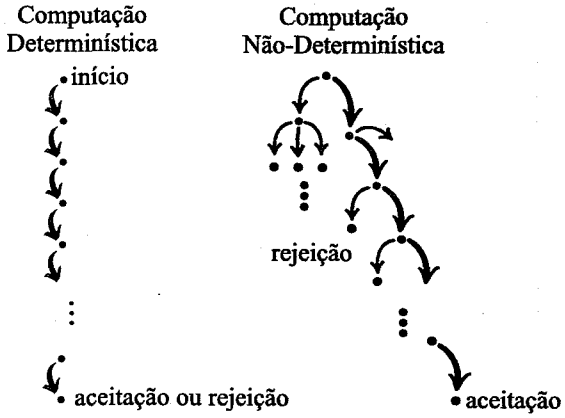


Figura 3.2: *Representação da computação em AFD e AFND [44].*

Definição Formal – Autômato Finito Não Determinístico

Um autômato finito não determinístico é uma 5-tupla $\langle Q, \Sigma, \delta, q_0, F \rangle$, em que:

1. Q é o *conjunto de estados*;
2. Σ é o *alfabeto*;
3. $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ é a *função de transição*;
4. $q_0 \in Q$ é o *estado inicial*; e
5. $F \subseteq Q$ é o conjunto de *estados de aceitação.*

De acordo com a definição da função de transição, dado o símbolo lido da palavra e o estado atual do autômato, é possível que o autômato venha a assumir qualquer combinação de estados possíveis, ou seja, um dos elementos do conjunto potência dos estados do autômatos, denotado por $\mathcal{P}(Q)$.

A computação de uma palavra ω por um AFND N se dá como segue: inicialmente o AFND encontra-se no estado inicial. Ao ler o primeiro símbolo de ω, N passa a assumir um ou mais estados, de acordo com o indicado pela função de transição. A partir deste ponto, deve-se verificar em quais estados o autômato está e quais ele deve assumir tomando cada símbolo a ser lido de ω. Ou seja, isto significa que a computação se dá independentemente em cada estado que o autômato se encontra.

Para aceitar uma palavra, após a leitura do seu último símbolo, basta que pelo menos um dos estados atuais do AFND seja um dos estados de aceitação. Caso contrário, o autômato rejeita a palavra.

Exemplo 3.7: Seja a palavra $\omega = 1010$ e o autômato finito não determinístico $N = \langle Q, \Sigma, \delta, q_0, F \rangle$, em que $Q = \{q_1, q_2, q_3, q_4\}$, $\Sigma = \{0, 1\}$ e a função de transição, estado inicial e de aceitação indicados no diagrama de estados ilustrado na Figura 3.3. Verifique se este autômato aceita ou rejeita a palavra ω.

Inicialmente, o autômato N encontra-se no estado q_1. Ao ler o símbolo $\omega_1 = 1$, este autômato assume os estados q_1

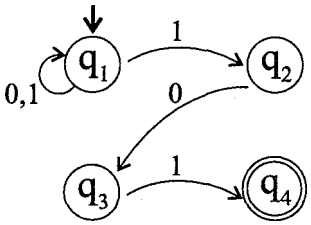


Figura 3.3: *Diagrama de Estados do AFND N*

$(\delta(q_1, 1) = q_1)$ e q_2 $(\delta(q_1, 1) = q_2)$. Ao ler o próximo símbolo $\omega_2 = 0$ este autômato assume os estados q_1 $(\delta(q_1, 0) = q_1)$ e q_3 $(\delta(q_2, 0) = q_3)$.

O terceiro símbolo a ser lido é $\omega_3 = 1$, que faz o autômato assumir os estados q_1 $(\delta(q_1, 1) = 1)$, q_2 $(\delta(q_1, 1) = q_2)$ e q_4 $(\delta(q_3, 1) = q_4)$. Por fim, ao ler o último símbolo $\omega_4 = 0$, o autômato assume os estados q_1 $(\delta(q_1, 0) = 1)$, q_3 $(\delta(q_2, 0) = q_3)$ e q_4 $(\delta(q_4, 0) = q_4)$.

Os estados que o autômato N assume durante a computação da palavra ω podem ser facilmente visualizados na Figura 3.4.

Visto que um dos estados que o autômato está após a leitura da palavra ω é o estado q_4, que está contido no conjunto de estados de aceitação, o autômato N aceita a palavra ω.

A representação de autômatos finitos não determinísticos também pode ser feita através de diagramas de estados e matrizes de transição. O diagrama de estados é obtido de forma análoga ao diagrama de estados dos AFD's, porém com a possibilidade de existir mais de um arco indicando transições a partir de um mesmo símbolo lido (ver Figura 3.3).

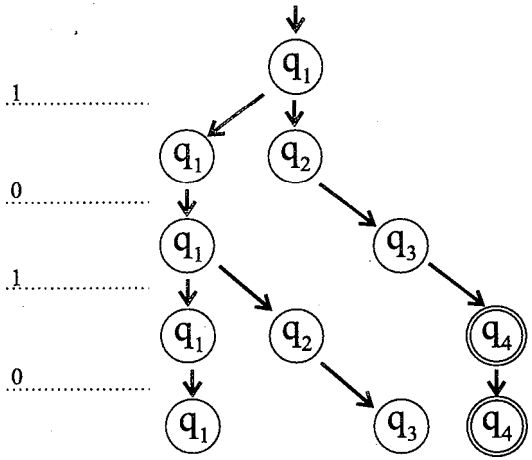


Figura 3.4: *Estados que o autômato N assume durante a computação da palavra ω.*

No caso das matrizes de transição, não há mudança em relação ao processo de construção indicado anteriormente para os AFD's, porém a aceitação de uma palavra pelo AFND é indicada por um valor inteiro e positivo.

Exemplo 3.8: Construa o diagrama de estados do AFND $N_2 = \langle Q, \Sigma, \delta, q_0, F\rangle$ em que $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{a, b\}$, $\delta(q_0, a) = q_0$, $\delta(q_0, a) = q_2$, $\delta(q_0, b) = q_1$, $\delta(q_2, b) = q_1$,$\delta(q_2, a) = q_0$ e $\{q_0, q_2\} \in F$ e verifique, através da utilização de matrizes de transição, se $\omega = aaa \in L(N_2)$.

O diagrama de estados do autômato N_2 pode ser obtido seguindo os passos para a representação de um AFD, porém

com o diferencial em relação à função de transição. O resultado pode ser visualizado na Figura 3.5.

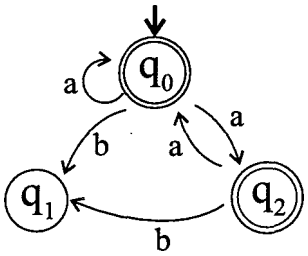


Figura 3.5: *Diagrama de estados do AFND* N_2.

Para verificar se $\omega \in L(N_2)$ é necessário construir as matrizes para efetuar a computação utilizando matrizes de transição. A matriz π que representa o estado inicial (q_0) é dada por:

$$\pi = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

A matriz η que representa os estados finais ($F = \{q_0, q_2\}$) é dada por:

$$\eta = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}.$$

As matrizes X_a e X_b correspondentes aos símbolos de Σ são dadas, respectivamente, por:

$$X_a = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} \qquad X_b = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Para verificar se ω é aceita por N_2 basta verificar o valor da seguinte expressão:

$$\begin{aligned} \pi^T \cdot X_{aaa} \cdot \eta &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}^3 \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 4 & 0 & 4 \\ 0 & 0 & 0 \\ 4 & 0 & 4 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \\ &= 8. \end{aligned}$$

Como 8 é um número inteiro e positivo, então pode-se concluir que ω é aceita pelo autômato N_2 e, conseqüentemente, $\omega \in L(N_2)$.

Apesar do não determinismo ser um diferencial destes autômatos em relação aos AFD's, a classe das linguagens reconhecidas por ambos é a mesma, ou seja, a classe das *linguagens regulares*.

No entanto, apesar de não serem mais abrangentes que os AFD's, os AFND's são úteis como ferramenta inicial na resolução de problemas envolvendo autômatos, pois, na maioria das

vezes, sua construção é mais rápida, seu funcionamento é mais simples de entender e são mais compactos (possuem menos estados) que seus equivalentes determinísticos [13].

Notas do Capítulo

Neste capítulo foram apresentados os modelos mais básicos de autômatos finitos e algumas de suas variantes.

Para aqueles que desejam expandir seus conhecimentos em autômatos finitos determinísticos e não-determinísticos, as obras de Newton Vieira [50], Paulo Blauth Menezes [29], Sipser [44], Hopcropt et al. [13] e Lewis & Papadimitrou [23] são excelentes pontos de partida, pois possuem uma linguagem bastante didática e apresentam exercícios diversos para melhor fixação do conteúdo. Além destas, as obras de Anderson [5] e Tao [47] ilustram diversas aplicações práticas dos autômatos finitos.

Exercícios Propostos

1. Projete um AF que controla um elevador num prédio de três andares. Descreva os estados, o alfabeto de entrada e a função de transição deste autômato. Com suas próprias palavras, explique o funcionamento deste autômato.

2. Um conjunto S é dito "fechado segundo uma operação" caso a operação aplicada sobre elementos de S sempre resulte em um elemento de S. Mostre que o conjunto das linguagens regulares é fechado sobre as operações a seguir:

(a) União;

(b) Concatenação;

(c) Estrela;

(d) Interseção;

(e) Complemento;

(f) Diferença.

3. Seja L uma linguagem regular. Mostre que existe um número infinito de autômatos finitos determinísticos que reconhecem L.

4. Sejam L_1 e L_2 duas linguagens formais. Prove ou refute a seguinte afirmação: "Se L_1 é uma linguagem regular e $L_2 \subseteq L_1$, então L_2 também é regular."

5. Consulte a obra *Introduction to the Theory of Computation* de Michael Sipser [44] e demonstre a equivalência entre AFND's e AFD's.

6. Mostre que a classe das linguagens regulares é um conjunto enumerável.

Capítulo 4

Autômatos Finitos Probabilísticos

Os *Autômatos Finitos Probabilísticos* (AFP's) não costumam ser objetos de estudo das disciplinas referentes às linguagens formais e autômatos dos cursos de graduação em Ciência da Computação. Porém, além de serem importantes para o entendimento de autômatos quânticos, estendem as aplicações dos autômatos finitos e são utilizados para diversas finalidades, como o reconhecimento de padrões e a aprendizagem de máquina. Uma gama de aplicações práticas pode ser obtida através da modelagem por meio de AFP's, por exemplo: organização automática de documentos, processamento de imagens, correção de texto corrompido, reconhecimento de cadeias de DNA, da escrita cursiva à mão, de voz, etc [24, 28].

4.1 Breve Introdução

O autômato finito probabilístico é uma generalização do autômato finito não-determinístico, onde uma probabilidade é atribuída a toda possível transição. Por exemplo, se em um AFND $\delta(q, \sigma) = \{q_1, q_2, \ldots, q_n\}$, então em um AFP, $p_i(q, \sigma)$ seria de-

finido como a probabilidade de o autômato passar do estado q para o estado q_i ao ler o símbolo σ.

Desse modo, a definição de um AFP implica na definição de uma *distribuição de probabilidades*. Como se sabe, uma distribuição de probabilidades descreve as chances que uma variável tem de assumir esse ou aquele valor ao longo de um espaço de valores. Nos AFP's, a variável é o estado que o autômato irá assumir e o espaço de valores é o conjunto de estados do autômato. Assim, as probabilidades relacionam-se aos estados que o autômato pode assumir sempre que uma transição ocorre, atuando de modo similar a uma escolha ponderada: as transições associadas aos maiores valores possuem maiores chances de serem efetuadas [38, 45].

A definição formal de um AFP é dada por:

Definição Formal – Autômato Finito Probabilístico

Um autômato finito probabilístico é uma 5-tupla $\langle Q, \Sigma, \delta, q_0, F \rangle$, em que:

1. Q é o *conjunto de estados*;
2. Σ é o *alfabeto*;
3. $\delta : Q \times \Sigma \times Q \rightarrow [0,1]$ é a *função de transição*;
4. $q_0 \in Q$ é o estado inicial; e
5. $F \subseteq Q$ é um conjunto de *estados de aceitação.*

A seguinte restrição em relação à função de transição deve ser respeitada:

$$\forall \sigma \in \Sigma^*, \forall q \in Q : \sum_{q' \in Q} \delta(q, \sigma, q') = 1.$$

Na computação de uma palavra $\omega = \omega_{(1)}\omega_{(2)}\ldots\omega_{(n)}$, um AFP A inicia o processamento no estado q_0 e lê o primeiro símbolo da palavra $(\omega_{(1)})$. O próximo estado de A irá depender da probabilidade definida na função de transição. Isto significa que A poderá assumir qualquer estado cuja probabilidade de transição seja não-nula (os estados q' tal que $\sum_{q' \in Q} \delta(q_0, \omega_{(1)}, q') \neq 0$).

O autômato prossegue dessa forma até que o último símbolo ($\omega_{(n)}$) seja lido.

É possível representar AFP's segundo matrizes de transição, de forma análoga aos modelos de autômatos já apresentados, mas com algumas mudanças:

- As matrizes relativas a cada símbolo possuem valores no intervalo $[0, 1]$, indicando as probabilidades associadas às transições;
- O somatório de cada linha das matrizes relativas aos símbolos deve ser igual a 1, respeitando a restrição apresentada na definição formal.

Estas duas alterações implicam, portanto, que as matrizes devem ser *matrizes estocásticas* e que o valor resultante do produto $\pi^T \cdot X_\omega \cdot \eta$ pertence ao intervalo $[0, 1]$.

Assim, a probabilidade do autômato A terminar o processamento da palavra ω em um estado de aceitação é definida por:

$$p_A(\omega) = \pi^T \cdot X_\omega \cdot \eta$$

em que $X_\omega = X_{\omega_{(1)}} \cdot \ldots \cdot X_{\omega_{(n)}}$.

Exemplo 4.1: Seja A o AFP cujo diagrama de estados está ilustrado na Figura 4.1. A partir da utilização de matrizes de

transição, verifique qual a probabilidade de A terminar o processamento de $\omega = ab$ em um estado de aceitação.

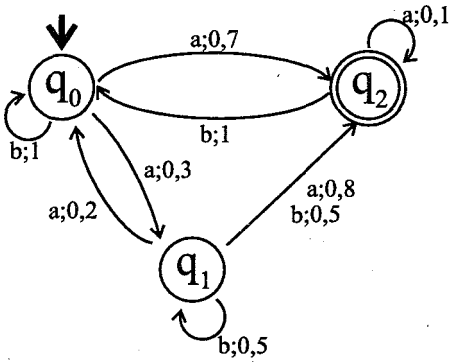


Figura 4.1: *Diagrama de estados do autômato finito probabilístico A.*

O estado inicial é q_0 e a matriz π dos estados iniciais é:

$$\pi = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Neste autômato existe apenas um estado final, $F = \{q_2\}$. Portanto, a matriz dos estados de aceitação η é:

$$\eta = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

A partir do diagrama é possível obter as matrizes X_a e X_b:

$$X_a = \begin{bmatrix} 0 & 0,3 & 0,7 \\ 0,2 & 0 & 0,8 \\ 0 & 0 & 1 \end{bmatrix} \qquad X_b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0,5 & 0,5 \\ 1 & 0 & 0 \end{bmatrix}$$

É importante salientar que as matrizes X_a e X_b respeitam a restrição imposta na definição formal dos AFP's. Para ilustração desta afirmação, basta verificar que o somatório das linhas das matrizes é igual a 1.

Para encontrar a probabilidade de A assumir um estado de aceitação ao ler ab, $p_A(ab)$, basta verificar o produto $\pi^T \cdot X_{ab} \cdot \eta$:

$$\begin{aligned} p_A(\omega) &= \pi^T \cdot X_{aba} \cdot \eta \\ &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0,7 & 0,15 & 0,15 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ &= 0,15. \end{aligned}$$

Ou seja, o autômato probabilístico A tem 15% de chance de terminar o processamento da palavra ab em um estado de aceitação.

4.2 Linguagens Estocásticas

Uma maneira natural de se definir a linguagem reconhecida por um AFP é através do conjunto de palavras cuja leitura leva o

autômato a um estado de aceitação com uma certa probabilidade mínima. Mas, qual seria essa probabilidade mínima? Essa probabilidade mínima é definida arbitrariamente e denomina-se *ponto de corte*.

Sejam A um AFP, λ um número real, $0 \leq \lambda < 1$, e $L(A, \lambda)$ o conjunto definido por:

$$L(A, \lambda) = \{\omega \in \Sigma^* : p_A(\omega) > \lambda\}$$

Se $\omega \in L(A, \lambda)$, diz-se que ω é *aceita* por A com ponto de corte λ. Então, $L(A, \lambda)$ é dita ser a linguagem de A com ponto de corte λ.

A classe das linguagens definidas por AFP's com ponto de corte é chamada *classe das linguagens estocásticas*.

Teorema 4.1: A classe das linguagens regulares é um subconjunto próprio da classe das linguagens estocásticas, ou seja, toda linguagem regular é estocástica e nem toda linguagem estocástica é regular.

Prova: Para provar este Teorema, deve-se provar as duas afirmações a seguir:

Teorema 4.1.1: Toda linguagem regular é estocástica.

Prova: AFD's podem ser considerados como um caso especial de AFP's. Se na definição de um AFD A $\delta(q, \sigma) = q_i$, então é

possível ver isto como se A entrasse no estado q_i com probabilidade 1. Assim, em re-escrevendo o AFD A como um AFP, a linha da matriz estocástica correspondente a $\delta(q,\sigma) = [q_0,\ldots,q_n]$ terá exatamente uma coordenada igual a 1 e todas as demais iguais a 0. Nesse caso, $p_A(\omega) = 1$, para $\omega \in \Sigma^*$ se, e somente se, $\omega \in L(A)$. Portanto, para qualquer λ, $0 \leq \lambda < 1$, tem-se que $L(A) = L(A,\lambda)$. Assim, todo a linguagem reconhecida por um AFD é trivialmente reconhecida por um AFP [38]. □

Teorema 4.1.2: Existem linguagens estocásticas que não são regulares.

Prova: Sejam $\Sigma = \{0,1\}$ e um AFP $A = \langle\{q_0,q_1\},\Sigma,\delta,q_0,\{q_1\}\rangle$ em que:

$$X_0 = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \qquad X_1 = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ 1 & 0 \end{bmatrix}$$

pode se verificar que, se

$$X_{\sigma_1}X_{\sigma_2}\ldots X_{\sigma_n} = \begin{bmatrix} m & p \\ q & r \end{bmatrix}$$

em que $\sigma_i \in \{0,1\}$, então $p = \cdot\sigma_n\sigma_{n-1}\ldots\sigma_1$, em que p está denotado como uma expansão binária.

Se $\omega = \sigma_1\sigma_2\ldots\sigma_n \in \Sigma^*$, então, como foi visto acima, $p_A(\omega) = 0.\sigma_n\sigma_{n-1}\ldots\sigma_1$. Os valores $p_A(\omega)$ são densos no intervalo fechado $[0,1]$. Isto implica que, se $0 \leq \lambda < \lambda_1 < 1$,

então $L(A, \lambda_1) \subset L(A, \lambda)$. Os conjuntos $L(A, \lambda)$, $0 \leq \lambda < 1$, formam, portanto, um conjunto não-enumerável. Porém, existe apenas um número enumerável de linguagens regulares. Portanto, existe λ tal que $L(A, \lambda)$ não é uma linguagem regular [38]. □

Teorema 4.2: Toda linguagem aceita por um AFP com ponto de corte $\lambda = 0$ é uma linguagem regular.

Prova: Seja $A = \langle Q_A, \Sigma, \delta_A, q_{0_A}, F_A \rangle$ um AFP que reconhece a linguagem $L(A, 0)$. Deve-se construir um AFND N, de modo que ele aceite toda palavra $\omega \in \Sigma^*$ que leva o AFP A do estado q_0 para um estado de aceitação, com probabilidade $p_A(\omega) > 0$, ou seja, que reconheça a mesma linguagem que o AFP A. Seja $N \langle Q, \Sigma, \delta, q_0, F \rangle$, em que $Q = Q_A$, $q_0 = q_{0_A}$, $F = F_A$ e δ construído da seguinte forma: para todo $j, l \in Q$ e $\sigma \in \Sigma$, se $p_{jl} \in X_\sigma$, e $p_{jl} > 0$, então $q_l \in \delta(q_j, \sigma)$ [22]. Não é difícil mostrar que $L(N) = L(A, 0)$ (a prova é solicitada na seção de Exercícios Propostos). □

Como foi visto, todas linguagem regular é estocástica, mas nem toda linguagem estocástica é regular. Porém, a linguagem $L(A, 0)$, em que A é um AFP qualquer, é uma linguagem regular.

Ainda é um problema em aberto saber se a classe das lingua-

gens estocásticas é fechada pelas operações booleanas (união, intersecção e complemento). Porém, sabe-se que esta classe é fechada pela operação reverso (a prova é solicitada na Seção de Exercícios Propostos no final deste capítulo). Considerando AFP's e AFD's, a linguagem formada pela união, intersecção e complemento das linguagens reconhecidas por estes autômatos é uma linguagem estocástica.

Teorema 4.3: Se L_R é uma linguagem regular e L_E é uma linguagem estocástica, então $L_R \cap L_E$, $L_R \cup L_E$ e $L_E - L_R$ são linguagens estocásticas.

Prova: Sejam L_E uma linguagem estocástica reconhecida por um AFP A com ponto de corte λ; L_R uma linguagem regular reconhecida por um AFD D; e $\omega \in \Sigma^*$.

1. A intersecção $L_R \cap L_E$ resulta em uma linguagem estocástica, uma vez que $p_A(\omega) \cdot p_D(\omega) > \lambda$ se, e somente se, $p_A(\omega) > \lambda$ e $p_D(\omega) = 1$;

2. A operação de complemento, pode ser denotada por meio da intersecção, ou seja, $L_E - L_R \equiv L_E \cap \overline{L_R}$ que foi demonstrada ser uma linguagem estocástica [35];

3. A prova da afirmação relativa à união é solicitada na seção de Exercícios Propostos. □

4.3 Ponto de Corte Isolado

Para assegurar uma *margem de erro* na probabilidade de aceitação de palavras por um AFP, define-se a aceitação com *ponto de corte isolado* (ou por *erro limitado*). Um AFP A reconhece uma linguagem L com ponto de corte isolado se existir uma margem de erro $\epsilon > 0$ tal que, para todo $\omega \in L$, $p_A(\omega) > \lambda + \epsilon$ e, para todo $\omega \notin L$, a probabilidade de A aceitar ω é $p_A(\omega) \leq \lambda - \epsilon$ [10].

Com esta restrição, a classe das linguagens aceitas por um AFP com ponto de corte isolado ($L(A, \lambda, \epsilon)$) é um subconjunto próprio da classe das linguagens regulares [38]. A prova desta afirmação é solicitada na seção de Exercícios Propostos.

Exemplo 4.2: Seja P um AFP cujo diagrama de estados está ilustrado na Figura 4.2. Verifique se este autômato aceita a palavra $\omega = aaa$ com ponto de corte isolado, em que $\lambda = 0,6$ e $\epsilon = 0,01$.

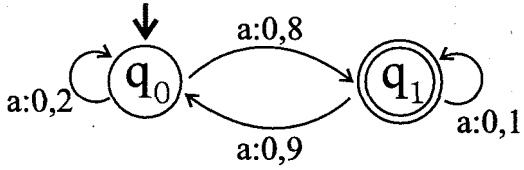


Figura 4.2: *Diagrama de estados do autômato finito probabilístico P.*

Para responder o problema em aberto do enunciado, basta construir as matrizes de transições com o propósito de obter o

valor da probabilidade de P aceitar ω. Assim:

$$\begin{aligned}
\pi^T \cdot X_{aaa} \cdot \eta &= \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0,2 & 0,8 \\ 0,9 & 0,1 \end{bmatrix}^3 \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} 0,368 & 0,632 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= 0,632
\end{aligned}$$

Uma vez que $p(\omega) = 0,632 > \lambda + \epsilon$, o autômato P aceita a palavra $\omega = aaa$.

Notas do Capítulo

Neste capítulo foi abordado o conceito de autômato finito probabilístico, cuja compreensão será essencial para o entendimento da versão quântica dos autômatos finitos a serem apresentadas a seguir. Para aqueles que desejam aprofundar os conhecimentos sobre os autômatos finitos probabilísticos e suas aplicações, recomenda-se a leitura dos livros de Paz [35] e de Levelt [22], bem como os artigos científicos de Stoelinga [45] e de Rabin [38]. Estas referências são um excelente material para aqueles que desejarem encontrar informações mais aprofundadas sobre AFP's.

Exercícios Propostos

1. Recorra ao Exemplo 3.1 e verifique se o AFP A aceita ou rejeita as seguintes palavras:

   (a) $\omega_1 = ba$, com ponto de corte $\lambda = 0,5$;

   (b) $\omega_2 = abaa$, com ponto de corte $\lambda = 0,6$ e $\epsilon = 0,05$;

   (c) $\omega_3 = a$, com ponto de corte $\lambda = 0,3$;

   (d) $\omega_4 = a$, com ponto de corte $\lambda = 0,7$;

   (e) $\omega_5 = a$, com ponto de corte $\lambda = 0,5$ e $\epsilon = 0.08$.

2. Recorrendo ao Exemplo 3.2, verifique se o AFP P aceita ou rejeita as seguintes palavras:

   (a) $\omega_1 = bbb$, com ponto de corte $\lambda = 0,6$;

   (b) $\omega_2 = aa$, com ponto de corte $\lambda = 0,6$ e $\epsilon = 0,01$;

   (c) $\omega_3 = abab$, com ponto de corte $\lambda = 0,9$;

   (d) $\omega_4 = abba$, com ponto de corte $\lambda = 0,4$;

   (e) $\omega_5 = a$, com ponto de corte $\lambda = 0,5$ e $\epsilon = 0.08$.

3. Consulte a obra de Levelt [22] e mostre que $L(N) = L(A, 0)$, em que N é um AFND construído a partir do AFP A com ponto de corte $\lambda = 0$..

4. Consulte a página 167 da obra *Introduction to Probabilistic Automata* [35] e prove que existe um AFP cuja linguagem que reconhece não é uma linguagem regular.

5. Prove que a classe das linguagens estocásticas é fechada sob a operação reverso. Para tanto, consulte a página 154 da obra de Paz [35].

6. Prove que a união de uma linguagem estocástica e uma linguagem regular é uma linguagem estocástica.

7. Consultando o Teorema da Redução, apresentado no artigo científico *Probabilistic Automata* de Michael Rabin [38], prove que há uma equivalência entre AFP's com ponto de corte isolado e AFD's.

8. Recorrendo ao Exemplo 3.2, se $\epsilon = 0,07$ o AFP P ainda aceitaria a palavra aaa?

Capítulo 5

Autômatos Finitos Quânticos

De acordo com Ambainis & Freivals [3], é possível que as primeiras implementações de computadores quânticos não sejam compostas inteiramente de componentes baseados na Mecânica Quântica. Ao invés disso, acredita-se que existirá uma parte quântica, uma parte clássica e uma forma de comunicação entre elas. Em um primeiro momento, em virtude da inovação e da baixa escalabilidade, a parte quântica será mais cara e mais reduzida que a parte clássica. Isto implica que serão necessários modelos computacionais que façam uso da Mecânica Quântica, mas que ao mesmo tempo sejam passíveis de implementação física.

Este possível cenário prático motiva o estudo dos *autômatos finitos quânticos* (AFQ's). Neste capítulo serão abordados os modelos fundamentais de AFQ's e algumas de suas propriedades referentes à classe das linguagens que reconhecem.

5.1 Breve Introdução

Autômatos Finitos Quânticos são modelos probabilísticos de computação cujos elementos são governados pelos postulados da Mecânica Quântica. Estruturalmente, um AFQ é composto por um conjunto de estados quânticos; uma fita clássica, na qual encontra-se a palavra de entrada; e um cabeçote clássico unidirecional, responsável pela leitura da palavra. Por serem sistemas quânticos, os estados podem estar em superposição.

A leitura de cada símbolo da palavra provoca a evolução do estado do autômato e, portanto, deve ser representada por meio de operadores unitários. Para se ter acesso ao resultado da computação da palavra, pelo menos uma medição do estado do autômato deve ser efetuada.

Na literatura a respeito, os modelos de AFQ's se diferenciam de várias formas, por vários critérios, entre os quais, o sentido de direção de leitura da palavra (unidirecional ou bidirecional) e o número de medições efetuadas (única ou múltiplas). Neste livro, serão considerados apenas os modelos unidirecionais (*1-way*), pois são os tipos de modelos de autômatos finitos geralmente estudados nos cursos de graduação em Ciência da Computação.

No estudo dos autômatos finitos quânticos unidirecionais, com relação ao número de medições efetuadas, existem duas variantes de interesse: *AFQ de Medição Única* e *AFQ de Múltiplas Medições*, que serão apresentadas nas seções a seguir.

5.2 AFQ de Medição Única

O *AFQ de Medição Única* (MO) foi o primeiro modelo de autômato finito quântico proposto na literatura científica [31]. Neste tipo de AFQ, uma medição é efetuada somente após a leitura de todos os símbolos da palavra.

A definição formal de um AFQ MO apresentada a seguir é uma versão equivalente à definição original [31], extraída de [9]:

Definição Formal – AFQ de Medição Única

Um autômato finito quântico de medição única é uma 5-tupla $\langle Q, \Sigma, \delta, |q_0\rangle, F\rangle$, em que:

1. Q é o *conjunto de estados* do autômato. Estes estados são vetores que formam uma base de um espaço de Hilbert n-dimensional;
2. Σ é um conjunto finito denominado *alfabeto*;
3. $\delta : Q \times \Sigma \times Q \to \mathbb{C}$ é a *função de transição*;
4. $|q_0\rangle \in Q$ é um vetor unitário de dimensão n que representa o *estado inicial* do autômato; e
5. $F \subseteq Q$ e define o subespaço vetorial de aceitação. Cada vetor $|q\rangle \in F$, define um projetor $|q\rangle\langle q|$. Então, o operador de projeção P_{ac} para o subespaço vetorial de aceitação é dado por: $P_{ac} = span\{|q\rangle \mid |q\rangle \in F\}$, ou seja, $P_{ac} = \sum_{|q\rangle \in F} |q\rangle\langle q|$.

Para todo $\sigma \in \Sigma$ e $|q_1\rangle, |q_2\rangle \in Q$, tem-se que:

$$\sum_{|q'\rangle \in Q} \overline{\delta(q_1, \sigma, q')} \cdot \delta(q_2, \sigma, q') = \begin{cases} 1, q_1 = q_2 \\ 0, q_1 \neq q_2 \end{cases}$$

O estado geral de um AFQ MO é representado por um vetor complexo n-dimensional e é denotado por:

$$|\psi\rangle = \sum_{i=0}^{n-1} \alpha_i |q_i\rangle$$

em que $n = |Q|$, $\{|q_i\rangle\}$ é o conjunto dos vetores da base correspondente aos estados do autômato, α_i é a amplitude do estado $|q_i\rangle$ e $\sum_{i=0}^{n-1} |\alpha_i|^2 = 1$.

Para cada símbolo σ lido, o autômato evolui de maneira unitária, de acordo com a seguinte equação:

$$U_\sigma |\psi\rangle = \sum_{|q_i\rangle, |q_j\rangle \in Q} \alpha_i \delta(q_i, \sigma, q_j) |q_j\rangle$$

em que U_σ é um operador unitário representado por uma matriz unitária.

A computação de uma palavra $\omega = \omega_{(1)} \ldots \omega_{(n)}$ por um AFQ MO A se dá da seguinte forma: o autômato inicia a computação da palavra no estado $|q_0\rangle$ e à medida que cada símbolo $\omega_{(i)}$, $1 \leq i \leq n$, é lido, aplica-se ao estado do autômato a matriz unitária $U_{\omega_{(i)}}$ correspondente. Este procedimento é aplicado sucessivamente a todos os símbolos da palavra ω. Ao final da leitura da palavra, aplica-se o operador de projeção P_{ac} para o subespaço vetorial de aceitação, ou seja, efetua-se uma medição do estado do autômato. Se o estado de A depois de ler a palavra ω é $|\psi_\omega\rangle$, então a probabilidade do autômato ser encontrado em um estado de aceitação é dado por:

$$\begin{aligned} p_A(\omega) &= \langle\psi_\omega| P_{ac} |\psi_\omega\rangle \\ &= ||P_{ac} |\psi_\omega\rangle||^2 \\ &= ||P_{ac} U_\omega |q_0\rangle||^2 \end{aligned}$$

Exemplo 5.1: Seja o autômato AFQ MO A, tal que

$$A = \langle \{|q_0\rangle, |q_1\rangle\}, \{0,1\}, \delta, |q_0\rangle, \{|q_0\rangle\} \rangle$$

em que δ é dada pelas matrizes de transições definidas na Tabela 5.1. Calcule a probabilidade do autômato A ser encontrado em um estado de aceitação ao ler a palavra $\omega = 010$.

Tabela 5.1: Transições do Autômato A

Transições
$U_0 \|q_0\rangle = \frac{1}{\sqrt{2}} \|q_0\rangle + \frac{1}{2} \|q_1\rangle$
$U_0 \|q_1\rangle = \frac{1}{\sqrt{2}} \|q_0\rangle - \frac{1}{2} \|q_1\rangle$
$U_1 \|q_0\rangle = \|q_1\rangle$
$U_1 \|q_1\rangle = \|q_0\rangle$

Inicialmente, o autômato está no estado $|q_0\rangle$ e lê o primeiro símbolo da palavra $\omega_{(1)} = 0$, portanto, realiza-se a transição $U_0 |q_0\rangle$, ou seja, o autômato assume o estado:

$$U_0 |q_0\rangle = \frac{1}{\sqrt{2}} |q_0\rangle + \frac{1}{\sqrt{2}} |q_1\rangle$$

O estado assumido pelo autômato após a leitura do primeiro símbolo é uma superposição igualmente distribuída dos estados $|q_0\rangle$ e $|q_1\rangle$.

Logo em seguida, A lê o segundo símbolo $\omega_{(2)} = 1$ e assume o estado:

$$U_1 \left(\frac{1}{\sqrt{2}} |q_0\rangle + \frac{1}{\sqrt{2}} |q_1\rangle \right) = \frac{1}{\sqrt{2}} |q_1\rangle + \frac{1}{\sqrt{2}} |q_0\rangle$$

Então, o autômato lê o último símbolo de ω ($\omega_{(3)}$):

$$\begin{aligned} U_0 \left(\frac{1}{\sqrt{2}} |q_1\rangle + \frac{1}{\sqrt{2}} |q_0\rangle \right) &= \frac{1}{\sqrt{2}} \left(\frac{1}{\sqrt{2}} |q_0\rangle + \frac{1}{\sqrt{2}} |q_1\rangle \right) + \\ &+ \frac{1}{\sqrt{2}} \left(\frac{1}{\sqrt{2}} |q_0\rangle - \frac{1}{\sqrt{2}} |q_1\rangle \right) \\ &= \frac{1}{2} |q_0\rangle + \frac{1}{2} |q_1\rangle + \frac{1}{2} |q_0\rangle - \frac{1}{2} |q_1\rangle \\ &= |q_0\rangle \end{aligned}$$

Após a leitura do último símbolo de ω, o autômato aplica o operador de projeção P_{ac} ao estado $|q_0\rangle$ e tem-se então:

$$\begin{aligned} p_A(010) &= \langle q_0 | P_{ac} | q_0 \rangle \\ &= \langle q_0 | q_0 \rangle \langle q_0 | q_0 \rangle \\ &= 1. \end{aligned}$$

Resumindo os passos efetuados:

$$\begin{aligned} p_A(\omega) &= ||P_{ac}U_{010}|q_0\rangle||^2 \\ &= |||P_{ac}|q_0\rangle||^2 \\ &= |||q_0\rangle\langle q_0|q_0\rangle||^2 \\ &= |||q_0\rangle||^2 \\ &= 1 \end{aligned}$$

Assim, a probabilidade do autômato A encontrar-se em um estado de aceitação após ler a palavra $\omega = 010$ é igual a 1.

5.3 AFQ de Múltiplas Medições

Kondacs & Watrous [21] introduziram o *Autômato Finito Quântico de Múltiplas Medições* (MM) no qual uma medição é efetuada após a leitura de cada símbolo da palavra de entrada.

A definição formal de um AFQ MM é dada a seguir:

Definição Formal – AFQ de Múltipla Medição

Um autômato finito quântico de múltipla medição é uma 6-tupla $\langle Q, \Sigma, \delta, |q_0\rangle, Q_{ac}, Q_{rej}\rangle$, em que:

1. Q é um conjunto finito de *estados*;
2. Σ é um conjunto finito denominado *alfabeto* o qual contém um marcador de término (\$);
3. $\delta : Q \times \Sigma \times Q \rightarrow \mathbb{C}$ é uma *função de transição* unitária, tal como definida para os AFQ's MO;
4. $|q_0\rangle \in Q$ é o estado inicial do autômato;
5. $Q_{ac} \subseteq Q$ é o conjunto de estados de aceitação; e
6. $Q_{rej} \subseteq Q$ é o conjunto de estados de rejeição.

Considera-se que os estados pertencentes a Q_{ac} ou a Q_{rej} são *estados de parada* e todos os demais estados são *estados de não-parada*, ou seja $Q_{non} = Q - (Q_{ac} \cup Q_{rej})$. Assume-se que $|q_0\rangle$ é um estado de não-parada e que Q_{ac} e Q_{rej} são conjuntos disjuntos, ou seja, $Q_{ac} \cap Q_{rej} = \emptyset$.

A definição formal dos AFQ's MM é bastante similar à dos AFQ's MO, por exemplo, em ambos os modelos a função de transição δ pode ser representada por matrizes unitárias. Porém, existem duas diferenças significativas: a existência

da transformação unitária $U_\$$, que deve ser efetuada quando o autômato efetuar a leitura do marcador de término da palavra; e a existência de estados de rejeição, que permitem verificar se a palavra lida pode vir a ser rejeitada, mesmo que todos os seus símbolos não tenham sido lidos pelo autômato.

A computação de uma palavra ω por um AFQ MM A é feita da seguinte forma: A inicia a computação no estado $|\psi\rangle = |q_0\rangle$. A transformação correspondente à leitura de um símbolo $\sigma \in \Sigma$ consiste em dois passos:

1. Primeiro, U_σ é aplicado ao estado do autômato. O novo estado $|\psi'\rangle$ será igual a $U_\sigma |\psi\rangle$;

2. Então, o estado $|\psi'\rangle$ é projetado com respeito a P_{ac}, P_{rej} e P_{non}, tais que:

$$\begin{aligned} P_{ac} &= span\{|q\rangle \in Q_{ac}\} \\ P_{rej} &= span\{|q\rangle \in Q_{rej}\} \\ P_{non} &= span\{|q\rangle \in Q_{non}\} \end{aligned}$$

   Após a projeção, existem três situações possíveis:

   (a) O autômato pára em um estado pertencente a Q_{ac}, com probabilidade dada pela amplitude $||P_{ac}|\psi'\rangle||^2$;

   (b) O autômato pára em um estado pertencente a Q_{rej}, com probabilidade dada pela amplitude $||P_{rej}|\psi'\rangle||^2$;

   (c) Com probabilidade $||P_{non}|\psi'\rangle||^2$, a computação continua e a próxima transformação, se houver, é aplicada.

No caso em que a computação continua, deve-se acumular as probabilidades de aceitação e rejeição observadas até então. A partir daí, o processo relativo a ler um símbolo da palavra, aplicar a transformação unitária correspondente e efetuar uma projeção segue até o autômato entrar num estado de parada ou até o fim da palavra. Caso o último símbolo da palavra seja lido e o autômato não tenha atingindo ainda um estado de parada, então aplica-se o operador $U_\$$, responsável por projetar a superposição em um dos subespaços de parada.

Exemplo 5.2: Seja o AFQ MM B em que:

$$\begin{aligned} Q &= \{|q_0\rangle, |q_1\rangle, |q_{ac}\rangle, |q_{rej}\rangle\} \\ \Sigma &= \{0\} \\ Q_{ac} &= \{|q_{ac}\rangle\} \\ Q_{rej} &= \{|q_{rej}\rangle\} \end{aligned}$$

em que $|q_0\rangle$ é o estado inicial e cuja função de transição está representada pelos operadores unitários descritos na Tabela 5.2. Determinar a probabilidade deste autômato parar em um estado de aceitação ao ler a palavra 00[1]

O primeiro passo é aplicar o operador U_0, referente ao símbolo lido ($\omega_{(1)} = 0$), ao estado inicial ($|q_0\rangle$):

[1]Exemplo adaptado de Isidro [16].

Tabela 5.2: Transições do autômato finito quântico de múltiplas medições B.

Transições
$U_0 \|q_0\rangle = \frac{1}{2}\|q_0\rangle + \frac{1}{2}\|q_1\rangle + \frac{1}{\sqrt{2}}\|q_{rej}\rangle$
$U_0 \|q_1\rangle = \frac{1}{2}\|q_0\rangle + \frac{1}{\sqrt{2}}\|q_1\rangle - \frac{1}{2}\|q_{rej}\rangle$
$U_\$ \|q_0\rangle = \|q_{rej}\rangle$
$U_\$ \|q_1\rangle = \|q_{ac}\rangle$

$$U_0 |q_0\rangle = \frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle + \frac{1}{2}|q_{rej}\rangle$$

Observa-se que após a leitura do primeiro símbolo da palavra houve uma superposição dos estados $|q_0\rangle$, $|q_1\rangle$ e $|q_{rej}\rangle$. Como B é um autômato de múltiplas medições, após a leitura do primeiro símbolo uma projeção deve ser efetuada.

Com probabilidade $\left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$ o estado de rejeição $|q_{rej}\rangle$ é observado. Neste caso, o valor $\frac{1}{2}$ é acumulado na probabilidade total de rejeição p_{rej}.

Com probabilidade $1 - \frac{1}{2} = \frac{1}{2}$, estados de não-parada são observados. Diante disto, a computação da palavra continua e o estado do autômato colapsa para o estado $\frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle$.

O próximo símbolo de ω a ser lido pelo autômato é $\omega_{(2)} = 0$. Aplicando-se novamente o operador U_0:

$$\begin{aligned} U_0\left(\frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle\right) &= \frac{1}{2}V_0|q_0\rangle + \frac{1}{2}V_0|q_1\rangle \\ &= \frac{1}{2}\cdot\left(\frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle + \frac{1}{\sqrt{2}}|q_{rej}\rangle\right) + \\ &+ \frac{1}{2}\cdot\left(\frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle - \frac{1}{\sqrt{2}}|q_{rej}\rangle\right) \\ &= \frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle \end{aligned}$$

Após a aplicação do operador U_0 efetua-se uma nova projeção. Como resultado, observa-se um estado de não-parada com probabilidade igual a 1, visto que na superposição resultante não existem estados de parada.

Como $|\omega| = 2$, já foi efetuada a leitura de todos os símbolos da palavra. Os próximos passos consistem em aplicar o operador $U_\$$ e efetuar uma medição:

$$\begin{aligned} U_\$\left(\frac{1}{2}|q_0\rangle + \frac{1}{2}|q_1\rangle\right) &= \frac{1}{2}V_\$|q_0\rangle + \frac{1}{2}V_\$|q_1\rangle \\ &= \frac{1}{2}|q_{rej}\rangle + \frac{1}{2}|q_{ac}\rangle \end{aligned}$$

Observa-se, com probabilidade $\left|\frac{1}{2}\right|^2 = \frac{1}{4}$, o estado de rejeição $|q_{rej}\rangle$ e com a mesma probabilidade, o estado de aceitação. A probabilidade total de aceitação será então $p_{ac} = \frac{1}{4}$, enquanto que a probabilidade de rejeição total (acumulada) é de $p_{rej} = \frac{1}{2} + \frac{1}{4} = \frac{3}{4}$. Portanto, a probabilidade de parar em um estado de aceitação é $\frac{1}{4}$ e a probabilidade de parar em um estado de

rejeição é $\frac{3}{4}$.

5.4 Propriedades e Abrangência dos AFQ's MO e MM

Uma vez que os estados de autômatos finitos quânticos são extensões dos estados de autômatos finitos clássicos, e podem, portanto, estar em superposição, poderia se pensar que seu poder computacional seria pelos menos igual aos dos seus análogos clássicos. Porém, dependendo do conceito de linguagem reconhecida utilizado (com ponto de corte ou com erro limitado), os modelos de autômatos finitos quânticos MO e MM são mais ou menos limitados com relação à classe das linguagens regulares ou de difícil caracterização a esse respeito. A seguir, são listadas as principais propriedades desses modelos com relação à classe de linguagens reconhecidas e algumas das principais relações entre eles. Por sua complexidade matemática, apenas algumas das provas serão apresentadas, ficando as demais como exercícios propostos e dirigidos àqueles alunos que tenham interesse em se dedicar ao assunto.

Assim como para os AFP's, o conceito de linguagem reconhecida por um AFQ pode ser definido de duas maneiras distintas e que tem repercussão distinta na classe de linguagens reconhecidas:

1. Ponto de corte: Um AFQ A reconhece uma linguagem L

com *ponto de corte* λ ($L(A, \lambda)$) se, para todo $\omega \in L$, a probabilidade de A terminar a computação em um estado de aceitação for maior que λ;

2. Ponto de corte isolado (erro limitado): Um AFQ A' reconhece uma linguagem L' com *ponto de corte limitado* ($L(A', \lambda, \epsilon)$) se existir um $\epsilon > 0$ tal que, para todo $\omega \in L'$, a probabilidade de A' aceitar ω é maior que $\lambda + \epsilon$ [9].

5.4.1 Propriedades dos AFQ's MO

Teorema 5.1: A classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está contida propriamente na classe das linguagens regulares.
Prova: A prova é solicitada na seção de Exercícios Propostos ao final deste Capítulo.

Teorema 5.2: Toda linguagem reconhecida por um AFQ MO (por ponto de corte ou por erro limitado) pode ser reconhecida por um AFP (por ponto de corte ou por erro limitado).
Prova: A prova é solicitada na seção de Exercícios Propostos ao final deste Capítulo.

Os teoremas 4.1 e 4.2 asseguram que os AFP's são estritamente mais poderosos que os AFQ's MO.

Teorema 5.3: A classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está contida propriamente na classe das linguagens reconhecidas por ponto de corte pelos AFQ's MO.

Prova: A prova é solicitada na seção de Exercícios Propostos ao final deste Capítulo.

Teorema 5.4: Existem linguagens não-regulares que são reconhecidas por ponto de corte pelos AFQ's MO.

Prova: Seja a linguagem $L = \{\omega \in \{a,b\}^* \mid |\omega|_a = |\omega|_b\}$. Como se sabe, esta linguagem não é regular.

O AFQ MO $A = \langle Q, \Sigma, \delta, |q_0\rangle, F\rangle$, em que $Q = \{|q_0\rangle, |q_1\rangle\}$, $\Sigma = \{a,b\}$, $F = \{|q_0\rangle\}$ e δ é definida pelas matrizes de transição:

$$X_a = X_b^{-1} = \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}$$

em que α é uma fração irracional de π.

O AFQ MO A reconhece L. A verificação desta afirmação é solicitada na Seção de Exercícios Propostos.

5.4.2 Propriedades dos AFQ's MM

Teorema 5.5: Todo AFQ MO pode ser simulado exatamente por um AFQ MM.

Prova: Simular exatamente significa que as distribuições dos autômatos são iguais.

Seja M $= \langle Q, \Sigma, \delta, q_o, F \rangle$ um AFQ MO e seja M' $= \left\langle Q', \Sigma, \delta', q_o, Q'_{ac}, Q'_{rej} \right\rangle$ um AFQ MM em que $n = |Q|$, $Q' = Q \cup \{q_n, q_{n+1}, \ldots, q_{2 \cdot n-1}\}$ é o conjunto de estados e Q'_{ac} e Q'_{rej} são definidos como:

$$\begin{aligned} Q'_{ac} &= \{q_{n+i} \in Q' | q_i \in F\} \\ Q'_{rej} &= \{q_{n+i} \in Q' | q_i \notin F\} \end{aligned}$$

O conjunto Q' consiste dos estados originais do autômato M acrescidos de um estado adicional para cada estado original. Seja δ representada por um conjunto de matrizes unitárias $\{U_\sigma\}_{\sigma \in \Sigma}$ e defina-se δ' pelo conjunto de matrizes $\left\{U'_\sigma\right\}_{\sigma \in \Sigma}$ em que a matriz U'_σ é:

$$U'_\sigma = \begin{bmatrix} U_\sigma & \\ & \mathbb{I}_n \end{bmatrix}$$

e

$$U'_\$ = \begin{bmatrix} U_\$ & \\ & \mathbb{I}_n \end{bmatrix} \cdot U_{flip}$$

em que

$$U_{flip} = \begin{bmatrix} & \mathbb{I}_n \\ \mathbb{I}_n & \end{bmatrix}$$

A matriz U_{flip} troca todas as amplitudes entre os estados em Q e seus correspondentes em $Q' \backslash Q$. Assim, M' tem probabilidade nula de parar até que o marcador de término de palavra tenha sido atingido e possui distribuição idêntica a M quando este pára. Assim, M' simula M exatamente. □

Teorema 5.6: A classe das linguagens reconhecidas por erro limitado por AFQ's MM está contida propriamente na classe das linguagens regulares.

Prova: A prova é solicitada na seção de Exercícios Propostos ao final deste Capítulo.

Teorema 5.7: A classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está contida na classe das linguagens reconhecidas por erro limitado pelos AFQ's MM.

Prova: Segue do Teorema 4.5. □

Teorema 5.8: A classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está propriamente contida na classe das linguagens reconhecidas por erro limitado pelos AFQ's MM.
Prova: A prova é solicitada na seção de Exercícios Propostos ao final deste Capítulo.

Teorema 5.9: A classe das linguagens reconhecidas por ponto de corte pelos AFQ's MO está contida na classe das linguagens reconhecidas por ponto de corte pelos AFQ's MM.
Prova: Segue do Teorema 4.5. □

Teorema 5.10: Os AFQ's MM são estritamente mais poderosos que os AFQ's MO.
Prova: Segue dos Teoremas 4.7, 4.8 e 4.9. □

5.5 AFQ Ancilla

O AFQ Ancilla foi proposto por Paschen [34] na tentativa de encontrar uma definição formal de AFQ equivalente ao AFD clássico.

A definição formal de um AFQ Ancilla é dada por:

Definição Formal – Autômato Finito Quântico Ancilla

Um AFQ Ancilla é uma 6-tupla $\langle Q, \Sigma, \Omega, \delta, |q_0\rangle, F\rangle$, em que:

1. Q é um *conjunto de estados*;
2. Σ é um *alfabeto de entrada*;
3. Ω é um *alfabeto da fita auxiliar*;
4. $\delta : Q \times \Sigma \times Q \times \Sigma \times \Omega \to \mathbb{C}[0,1]$ é uma *função de transição*;
5. $|q_0\rangle$ é o *estado inicial*;
6. $F \subseteq Q$ é um conjunto de *estados de aceitação.*

A função de transição δ obedece a seguinte restrição $\forall \sigma \in \Sigma$ e $q_1, q_2 \in Q$:

$$\sum_{q \in Q, \gamma \in \Omega} \overline{\delta(q_1, \sigma, q, \gamma)} \delta(q_2, \sigma, q, \gamma) = \begin{cases} 1, & q_1 = q_2 \\ 0, & q_1 = q_2 \end{cases}$$

A computação de uma palavra ω por um AFQ Ancilla A se

dá de modo similar à computação de um AFQ MO, diferenciando-se apenas pela função de transição: no caso dos AFQ's Ancilla, a função de transição leva em consideração uma fita auxiliar na qual escreve-se após cada transição, mas que nunca se lê desta fita – é deste fato que advém o nome do autômato, pois *ancilla* significa *auxiliar*. No AFQ Ancilla, tal como no AFQ MO, a medição é efetuada apenas ao final da leitura do último símbolo da palavra [16].

O AFQ Ancilla é, no mínimo, tão "poderoso" quanto os modelos de AFD e AFND, apresentados no Capítulo 3. Paschen mostra que dado um AFD A mínimo que reconhece uma linguagem L, é possível construir um AFQ Ancilla B que reconhece L com probabilidade igual a 1 [34].

5.6 Outros Modelos

Os modelos apresentados ao longo deste Capítulo são as variantes básicas dos AFQ's. Na literatura existem outras definições possíveis. Algumas destas definições são:

- AFQ Multifita: Proposto por Ambainis et al. como sendo uma generalização do modelo AFQ MM acrescido de múltiplas fitas. Foi provado que existe uma linguagem reconhecida por tal modelo que não é reconhecida por autômato finito determinístico ou probabilístico [2];

- AFQ Bidirecional de Múltiplas Medições: Similar ao AFQ MM apresentado na Seção 5.3, porém a palavra de en-

trada pode ser lida no sentido normal, da esquerda para a direita, como também da direita para a esquerda [21];

- AFQ Bidirecional com Estados Clássicos e Quânticos: é uma variação do modelo 2-way AFQ de Múltiplas Medições cujos estados podem incluir estados quânticos, mas cuja movimentação de leitura é requerida ser clássica [4];

- AFQ Unidirecional com Linguagem de Controle: Utiliza uma linguagem regular auxiliar, denominada linguagem de controle. Este modelo permite observações após cada símbolo lido, mas a aceitação da palavra só ocorre se o resultado da computação pertencer à linguagem de controle. As linguagens aceitas com ponto de corte isolado por este autômato são linguagens regulares [8].

Notas do Capítulo

Este capítulo apresentou a definição de autômatos finitos quânticos por meio de três variantes básicas distintas. Por meio do estudos de tais variantes foi possível conhecer como se dá a computação de uma palavra e também como estes modelos se relacionam, em termos da linguagem reconhecida, com seus equivalentes clássicos.

Para aqueles que desejam expandir os conhecimentos sobre este tema, recomenda-se como ponto de partida a leitura das dissertações de Brodsky [9] e Isidro [16], esta última em língua portuguesa. Além destes, recomenda-se os artigos de Moore

& Crutchfield [31], Kondacs e Watrous [21] e Paschen [34] que discutem os modelos de autômatos quânticos que apresentam.

Exercícios Propostos

1. Utilizando o AFQ MO A definido no Exemplo 4.1, verifique se as seguintes palavras pertencem a $L(A)$:

   (a) $\omega_2 = 0101$, com ponto de corte $\lambda_2 = 0.5$;

   (b) $\omega_3 = 1100$, com ponto de corte limitado, $\lambda = 0.4$ e $\epsilon = 0.02$.

2. Utilizando o AFQ MM B definido no Exemplo 4.2, verifique se as seguintes palavras pertencem a $L(B)$:

   (a) $\omega_1 = 000$, com ponto de corte $\lambda_1 = 0.75$;

   (b) $\omega_2 = 0101$, com ponto de corte $\lambda_2 = 0.5$;

   (c) $\omega_3 = 1100$, com ponto de corte limitado, $\lambda = 0.4$ e $\epsilon = 0.02$.

3. Prove que a classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está contida propriamente na classe das linguagens regulares. Para tanto, consulte a página 19 da dissertação de Brodsky [9];

4. Consulte a página 6 do artigo *Characterizations of 1-Way Quantum Finite Automata* dos autores Brodsky & Pippenger [10] e prove que toda linguagem reconhecida por um AFQ MO (por ponto de corte ou por erro limitado)

pode ser reconhecida por um AFP (por ponto de corte ou por erro limitado);

5. Consulte o trabalho de Brodsky [9] e verifique que a classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está contida propriamente na classe das linguagens reconhecidas por ponto de corte pelos AFQ's MO;

6. Volte ao Teorema 4.4 e prove que o AFQ MO A reconhece a linguagem L;

7. Consulte a Proposição 6 apresentada no artigo *On the power of Quantum Finite State Automata* de Kondacs & Watrous e prove que a classe das linguagens reconhecidas por erro limitado por AFQ's MM está contida propriamente na classe das linguagens regulares;

8. Consulte a dissertação de Brodsky [9] para provar que (i) a classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está *contida* na classe das linguagens reconhecidas por erro limitado pelos AFQ's MM e que (ii) a classe das linguagens reconhecidas por erro limitado pelos AFQ's MO está *propriamente contida* na classe das linguagens reconhecidas por erro limitado pelos AFQ's MM;

9. Consulte o trabalho *Quantum Finite Automata Using Ancilla Qubits* de Paschen [34] e prove que dado um AFD A mínimo que reconhece uma linguagem L, é possível

Referências Bibliográficas

[1] Alfred V. Aho, Ravi Sethi, and Jeffrey D. Ullman. *Compilers: Principles, Techniques and Tools.* Addison-Wesley, 1986.

[2] Andris Ambainis, Richard F. Bonner, Rusins Freivalds, Marats Golovkins, and Marek Karpinski. Quantum Finite Multitape Automata. In Springer, editor, *SOFSEM '99, Theory and Practice of Informatics, 26th Conference on Current Trends in Theory and Practice of Informatics*, volume 175 of *Lecture Notes in Computer Science*, 1999.

[3] Andris Ambainis and Rusins Freivalds. 1-way quantum finite automata: strengths, weaknesses and generalizations. In *FOCS '98: Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, page 332, Washington, DC, USA, 1998. IEEE Computer Society.

[4] Andris Ambainis and JohnWatrous. Two-way Finite Automata with Quantum and Classical States. *Theoretical Computer Science*, 287:299–311, 2002.

[5] James A. Anderson. *Automata Theory with Modern Applications.* Cambridge University Press, 2006.

[6] Michael A. Arbib. *Brains, Machines and Mathematics.* Springer-Verlag, 1987.

[7] Daniela Berardi, Fabio De Rosa, Luca De Santis, and Massimo Mecella. Finite State Automata as Conceptual Model for -services. *Journal of Integrated Design and Process Science*, 8:105–121, 2004.

[8] Alberto Bertoni, Carlo Mereghetti, and Beatrice Palano. Quantum Computing: 1-way Quantum Automata. In *Developments in Language Theory, 7th International Conference, DLT 2003*, pages 1–20, 2003.

[9] Alexander Brodsky. Models and Characterizations of 1-way Quantum Finite Automata. Master's thesis, University of Waterloo, 1998.

[10] Alexander Brodsky and Nicholas Pippenger. Characterizations of 1-way Quantum Finite Automata. *SIAM Journal of Computing*, 31:1456–1478, 2002.

[11] Maxime Crochemore and Christophe Hancart. Automata for Matching Patterns. *Handbook of Formal Language*, 2:399–462, 1997.

[12] Paul Dirac. *The principles of Quantum Mechanics.* Oxford UK, 4th edition, 1982. ISBN 0198520115.

[13] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introdução à Teoria dos Autômatos, Linguagens e Computação.* Editora Campus, 2002.

[14] David A. Huffman. The Synthesis of Sequential Switching Circuits. *Journal of the Franklin Institute*, 257:164–190, 1954.

[15] Sandor Imre and Ferenc Balazs. *Quantum Computing and Communications - An Engineering Approach.* John Wiley & Sons, 2005.

[16] Cheyenne Ribeiro Guedes Isidro. Uma Abordagem Quântica para o Uso de Expressões Regulares. Master's thesis, Universidade Federal de Campina Grande, 2008.

[17] Lauri Karttunen. Applications of Finite-State Transducers in Natural Language Processing. In *5th International Conference of Implementation and Application of Automata*, pages 34–46, 2000.

[18] Phillp Kaye, Raymond Laflamme, and Michele Mosca. *An Introduction to Quantum Computing.* Oxford University, 2007.

[19] Arnolds Kikusts. A Small 1-way Quantum Finite Automata. Disponível para download em http://arxiv.org/quantph/9810065., 1998.

[20] Stephen C. Kleene. Representation of Events in Nerve Nets and Finite Automata. *Automata Studies*, 1:3–42, 1956.

[21] Attila Kondacs and John Watrous. On the Power of Quantum Finite State Automata. *38th Annual Symposium on Foundations of Computer Science – IEEE Computer Society*, 1:66–75, 1997.

[22] Willem J. M. Levelt. *An Introduction to the Theory of Formal Languages and Automata.* John Benjamins Publishing Company, 2008.

[23] Harry R. Lewis and Christos H. Papadimitriou. *Elementos de Teoria da Computação.* Bookman, 2004.

[24] Giuseppe Anthony Nascimento Lima. Autômatos Finitos Probabilísticos e o Reconhecimento de Padrões. Technical report, Departamento de Sistemas e Computação – Universidade Federal de Campina Grande, 2008.

[25] Warren McCulloch and Walter Pitts. A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, 1:115–133, 1943.

[26] David McMahon. *Quantum Computing Explained.* John Wiley & Sons, 2008.

[27] George H. Mealy. A Method for Synthesizing Sequential Circuits. *Bell Systems Technical Journal*, 34:1045–1079, 1955.

[28] Andréa Pereira Mendonça. Autômatos Finitos Probabilísticos e Aplicações em Aprendizagem de Máquina. Technical report, Departamento de Sistemas e Computação – Universidade Federal de Campina Grande, 2007.

[29] Paulo Blauth Menezes. *Linguagens Formais e Autômatos.* Editora Sagra Luzzatto, 1997.

[30] Mehryar Mohri. On Some Applications of Finite-State Automata Theory to Natural Language Processing. *Natural Language Engineering*, 2:61–80, 1996.

[31] Cristopher Moore and James P. Crutchfeld. Quantum automata and quantum grammars. *Theoretical Computer Science*, 237:275–306, 1997.

[32] Edward F. Moore. Gedanken Experiments on Sequential Machines. *Automata Studies*, 1:129–156, 1956.

[33] Michael A. Nielsen and Isaac L. Chuang. *Computação Quântica e Informação Quântica.* Bookman, 2005.

[34] Kathrin Paschen. Quantum Finite Automata Using Ancilla Qubits. Technical report, University of Karlsruhe, Maio 2000.

[35] Azaria Paz. *Introduction to Probabilistic Automata.* Academic Press, 1971.

[36] Dominique Perrin. Les Debuts de la Theorie des Automates. Technical report, Laboratoire d'Informatique Algorithmique: Fondements et Applications, 1993.

[37] Charles Petzold. Codes. Microsof Press, 1999.

[38] Michael O. Rabin. Probabilistic automata. *Information and Control*, 6:230–245, 1963.

[39] Michael O. Rabin and Dana Scott. Finite Automata and their Decision Problems. *IBM Journal of Research and Development*, 3:114–125, 1959.

[40] Emmanuel Roche and Yves Schabes. Finite-State Language Processing. The MIT Press, 1997.

[41] Dana Ron. *Automata Learning and its Applications.* PhD thesis, Hebrew University, 1995.

[42] Thomas Schwentick. Automata for XML – A Survey. *Journal of Computer and System Sciences*, 73:289–315, 2007.

[43] Charles C. Sims. *Computation with Finitely Presented Groups.* Cambridge University Press., 1994.

[44] Michael Sipser. *Introduction to the Theory of Computation.* PWS Publishing Company, 1997.

[45] Marielle Stoelinga. An introduction to probabilistic automata. *Bulletin of the European Association for Theoretical Computer Science*, 198:78–176, 2002.

[46] Andrews S. Tanenbaum. *Structured Computer Organization.* Amsterdam, The Netherlands, 2005.

[47] Renji Tao. *Finite Automata and Application to Cryptography.* Springer, 2009.

[48] Alan Turing. On Computable Numbers, With An Application to the Entscheidungsproblem. In *Proceedings of the London Mathematical Society*, 2, pages 230–267, 1936.

[49] Moshe Y. Vardi and PierreWolper. An Automata-Theoretic Approach to Automatic Program Verification. In *First Symposium on Logic in Computer Science*, pages 322–331, 1986.

[50] Newton José Vieira. *Introdução aos Fundamentos da Computação.* Pioneira Thomson, 2006.

[51] Fritz von Haeseler. *Automatic Sequences.* Walter de Gruyter, 2003.

[52] Colin P. Williams and Scott H. Clearwater. *Explorations in Quantum Computing.* The Eletronic Library of Science, 1998.

Índice Remissivo